PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

BRA00149

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
G06F 3/14

A1

(11) International Publication Number: WO 00/23872

(43) International Publication Date: 27 April 2000 (27.04.00)

(21) International Application Number: PCT/US99/24369

(22) International Filing Date: 18 October 1999 (18.10.99)

(30) Priority Data:
09/173,853 16 October 1998 (16.10.98) US

(63) Related by Continuation (CON) or Continuation-in-Part (CIP) to Earlier Application
US 09/173,853 (CON)
Filed on 16 October 1998 (16.10.98)

(71) Applicant *(for all designated States except US)*: PROPHET FINANCIAL SYSTEMS, INC. [US/US]; 430 Cambridge Avenue, Palo Alto, CA 94306 (US).

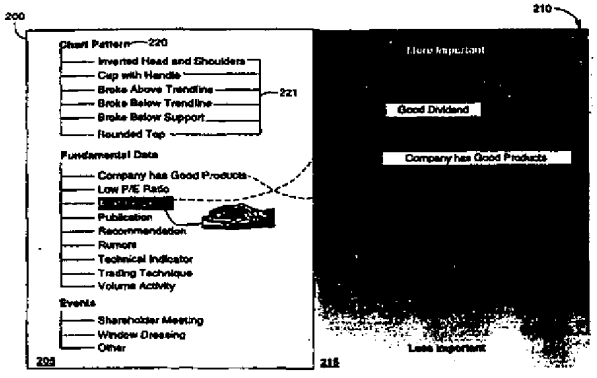
(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: KNIGHT, Timothy, O. [US/US]; 1171 Fife Avenue, Palo Alto, CA 94301 (US).

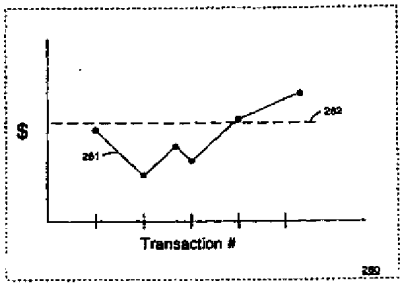
(74) Agent: GROSS, John, Nicholas; Suite B, 10950 N. Blaney, Cupertino, CA 95014 (US).

(81) Designated States: CA, CN, DE, GB, JP, MX, RU, US, Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: GRAPHICAL DATA COLLECTION INTERFACE





(57) Abstract

This is a graphical user interface which provides for capturing application data in picture form. A set of data capture tools, including a subjective data parameter palette and accompanying data canvas (215) are presented to an operator during an interactive session. By selecting and placing such data parameters (230) on the data canvas, a user can paint a data picture representing his/her subjective motivations, mental impressions, reasons, etc. for engaging in a particular transaction. In a preferred embodiment, the present interface is used in connection with a stock portfolio management application, which is used to capture user rationale and logic for purchasing or selling financial instruments such as securities, options, etc.

Graphical Data Collection Interface

FIELD OF INVENTION

The present invention relates to a graphical user data collection/presentation interface. The invention is particularly suited for use with computer programs intended as personal information assistants, consumer preference/opinion gathering tools, and in similar environments.

BACKGROUND OF INVENTION

User interfaces for computer programs and operating systems are well-known in the art. At first, such interfaces were entirely text based, and thus primitive, difficult to use, and lacking in functionality. This limitation, too, restricted their use primarily to a small segment of the population consisting of advanced skill computer users. With the advent of graphical interfaces by Xerox, Apple, Microsoft, etc., the use of computers has expanded dramatically to touch upon the entire potential consuming public. Furthermore, the use of graphical interfaces has improved the functionality of computers significantly, so that functions that once required numerous lengthy text based input parameters can now be performed by simple iconic replacements. For example, the task of copying a file from one system drive to another once required extensive knowledge of the exact syntax format of the system, combined with lengthy keyboard data entry. Now, such function can be performed simply by clicking a mouse button on a graphical representation of such file on the first drive, and then manually dragging and dropping such file onto another icon representing the second drive. Utility, productivity, etc., have all increased substantially now because tasks that once required numerous cumbersome operations can now be performed in a fraction of the time, and without lengthy operator training

procedures. This is due, in part, to the fact that most users can intuitively grasp the nature of a function when it is presented in visual form to mimic a real-life manual operation; in this case, the act of moving a file (represented in icon form to resemble a paper document) from one location (represented in icon form as a filing cabinet) to another (represented in icon form as a different filing cabinet). For a useful background reference on this topic, a reference by Schneiderman, entitled "Designing the User Interface," is recommended.

To date, nonetheless, graphical interfaces have been used in computer applications (those programs running under the operating system) primarily for processing only objective data items. For purposes of the present disclosure, a distinction is loosely made between data items that are *objective* - i.e., can be quantified by some known benchmark outside the user's mental/cognitive impressions - and *subjective*, i.e., those data items that are primarily based on the user's personal preferences. As an example, an objective data item might be the temperature at a particular location and time; this can be represented by a data value - i.e., some number in Centigrade - that can be identified/verified by another measurement tool. Similarly, the amount of money in one's bank account can be quantified numerically with a known denomination. In contemporary programming form, this collection of data from the user shows up in, among other places, personal financial planning programs, which might ask a user to identify the real rate of return (a % number) expected/desired by the user for his/her investments.

In contrast, a subjective data item could be the personal enjoyment an individual attains as a result of listening a particular piece of music, the importance they assign to one factor that is part of particular choice they make, etc. For instance, a typical person purchasing an automobile could rank the following in the order of importance in their selection of a particular model: price, performance, comfort, insurance, and so on. Similarly, when asked why a specific course of action was taken, such as selling an equity position in a company, an individual might identify that he/she thought the price had reached a subjective target value, or that the company was lagging its competitors, or that the local newspaper ran a negative article on the company, etc., etc.

It should be understood that these are but typical examples, and it is apparent that a multitude of similar situations arise each day in connection with a person's experiences/interactions with the world around them. The above, of course, is a very limited explanation of the general differences between subjective and objective data items, and, of course, should be understood as such. There may be a relationship between the two, as for example when an objective data item (the occurrence of an event such as the announcement of a new product by a company) affects and/or results in a subjective data item (a person deciding to buy the company's stock). Thus, this classification is intended merely as an illustrative tool for explaining the basis of the present invention, and should not be taken as a limitation on any of the teachings set forth herein.

Some typical reasons why contemporary programs do not handle such subjective information, include, of course, the fact that few such programs are able to translate this information into machine-manipulable form so that meaningful conclusions can be drawn from the same, and that such results can be conveyed in some intelligent fashion to the user. Fewer still are able to collect this data in an efficient, user-friendly manner; those that do collect subjective data items do so using relatively primitive techniques. For example, the same personal financial planning program described above might ask a user to identify the level of risk he/she is willing to accept, by placing an electronic check mark on a screen form listing such options as "High," "Medium," "Low," etc. Similarly, a conventional on-line purchasing/marketing form might ask the user to identify on a scale of 1 - 10 the importance of various features of a product. To receive the user's input, an electronic representation of a sliding scale might be presented to the user, which he/she can manipulate and set with a computer mouse to a particular value. This is one means of effectuating the graphical object - action interface described above, but is somewhat limited because the user is required to adopt and accept the graphical objects, tools, and options presented to express his/her choices.

The general problems associated with interfaces that attempt to extract individual subjective data items include the fact that: (1) they rely on non-graphical interfaces, which make them unattractive, unintuitive and unfriendly to prospective

users; (2) they present the user with a limited set of objects and interactions for his/her use; in other words, an online questionnaire for example might ask only about the four most important variables as perceived by the vendor, when in fact there may be a whole slew of factors important to the potential customer filling out such questionnaire; (3) they do not permit a user to ignore those items that are not of interest in the provided for universe of parameters; instead, they require the user to fill out page after page of questions, many of which may not be relevant or important to such user; (4) they take too much time to complete because they require cumbersome keyboard operations, and this results in poor data yield caused solely by user impatience; (5) they often require users to provide actual arithmetic or mathematical data input to represent data values perceived only subjectively by such users; in other words, if they ask a user to rate car characteristics, such person might have to assign a color parameter of a car as a "5," and a price parameter of such car as an "8". Later, the user might believe that the acceleration time is also important, and he/she would then be forced to compute some new value that is appropriate relative to the other numerical values already provided for other parameters. Furthermore, consideration of a new parameter might require scaling or re-adjustment of all prior values for other parameters. Such arithmetic manipulation is cumbersome and beyond the capability or interest level of many potential users of such programs.

It is apparent that many of these same problems are inherent in conventional objective data collection/presentation systems, to the extent they even utilize a graphical interface. Accordingly, such systems would benefit from a solution that ameliorates such problems.

SUMMARY OF THE INVENTION

The present invention, therefore, aims to provide a graphical interface which permits a user to select from a palette of preferences to describe those factors influencing his/her subjective opinion, actions, etc., relating to various items/events, and which allows such user to identify such factors to a data processing application efficiently, quickly, intuitively and without substantial human interaction;

An additional goal of the present invention is to provide a mechanism and

method for application users to express subjective data in graphical, rather than arithmetic form, so that data entry procedures are simplified and made more efficient;

A further object of the present invention is to provide an apparatus and method for performing meaningful analyses of subjective data, by permitting application users to express subjective data in a graphical form that is nevertheless translatable into a data form that can be handled mathematically by an underlying application program;

Yet another object of the present invention is to provide a mechanism and method to permit a user to express a relative ranking of parameters pertaining to a particular topic/event in visual form, without requiring extensive data input, or human interaction/analysis;

Another object of the present invention is to provide a mechanism and method for users to express personal preference data items, as well as relationships between such items, through graphical tools which visually depict and/or prioritize such data items in a manner conducive to quick and easy understanding by the user;

A further object of the present invention is to provide a system and method for users to identify, store, recall and modify experiences, lessons and understandings gleaned from participating in various actions and transactions, so that they may learn and benefit from their past mistakes and successes when they adopt particular strategies for engaging such actions and transactions;

A preferred embodiment of an electronic interface of the present invention achieves these objects by permitting a user to communicate subjective data information concerning a proposed or actual action/transaction (i.e., such the user's mental impressions of such event, an item of interest, or some lesson learned by such user associated with the same) to an underlying application program during what can be generally described as a "data painting sessions." The interface includes a parameter "menu," (or pallette) which menu provides a user with a visible set of data parameters which may be associated with the subjective data information. The parameters can be presented in text form, symbolic form, or some other form easily comprehendable by the user, and can be customized in part by the user to reflect individual preferences. In a separate portion of the interface a parameter "canvas," is presented so that it is simultaneously visible with the parameter menu. The user can

select and manipulate the data parameters, placing them on the parameter canvas using a drag and drop motion, thus generating a kind of data "picture" for the action/transaction. This data picture can be stored, retrieved, edited and modified at will during a later session if desired. The interface is preferably configured so that all of the user's subjective data information is captured using only the parameter menu and canvas tools presented by such interface, and during a single data collection session.

Again in a preferred embodiment, the interface is configured so that the data parameters associated with the subjective data information are selected and moved by such user from the parameter menu to the parameter canvas using an an electronic pointing device. The menu and canvass are located proximately to each other so that the user can perform the act of moving the parameters to the canvass in rapid fashion.

At the end of the data painting session in this preferred embodiment, the identified data parameters associated with the subjective data information are stored as one or more electronic records corresponding to an electronic data picture. Notably, this data picture includes numeric data values, but is generated transparently without numeric data input by the user, thus reducing the burden experienced by the user in presenting his/her data selections. The numeric data values are based on the physical location of the data parameters as placed by the user on the parameter canvas, thus allowing the user to rank the parameters in relative importance quickly, easily, and without computational effort on their part. This relative ranking between data parameters can be changed by the user by simply altering a relative physical arrangement of the data parameters on the parameter canvas using a computer mouse or similar pointing device. In certain applications for more skilled users, the data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

Further in the preferred embodiment, the parameter canvas includes a gradient visible to the user for assisting in the ranking of the data parameters. The gradient is presented in differing shades of color as reinforcing feedback. Additionally, while the user is arranging the data parameters, the parameter canvas conveys visible feedback information, alerting him/her to the action they are taking.

In another variation, and so as to save time and burden on the user, the interface can present an initial data picture to the user based on that person's prior data pictures. This initial data picture can then be modified as desired to reflect the particular action/transaction.

To further enhance the user's utilization of the interface, an additional chart or picture can be presented to them depicting various correlations between the data picture being created and prior data pictures made by such user. In this fashion, the user can determine at a glance what the outcome of a particular action/transaction is likely to be based on the variables identified at that time.

A preferred user data capture method of the present invention utilizes the aforementioned interface for permitting a user to identify personal parameters concerning an action and/or transaction to an underlying computer program.

Although the invention is described below in a preferred embodiment associated with a personal stock portfolio managing program, it will be apparent to those skilled in the art that the present invention would be beneficially used in many other applications where it is desirable to capture data in a fast, easy, and comprehensive manner that minimizes burden on a user providing such input.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a flow chart depicting the general operation of an embodiment of the present invention;

Figure 2 is a screen shot from a computer program application incorporating one embodiment of the interface of the present invention;

DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 is a flow chart illustrating the basic operation of a data collection interface portion 100 of an application program incorporating the present invention. In a preferred embodiment the present interface is used by a stock portfolio managing program to elicit feedback and information concerning a user's motivations, opinions, reasonings, etc. for participating in a particular transaction - i.e., by either purchasing or selling a particular equity. This program is entitled "TotalTrader" and can be

obtained from visiting a website maintained by the assignee at www.totaltrader.com.

As used herein, nonetheless, the terms "action" or "transaction" are intended in their broadest sense to mean an event of interest to the user that has already occurred or may occur in the future, or even an article or item of interest to the user. For instance, in the latter case, when the present interface is used in connection with an online marketing or sales research program, a transaction can represent a user's interest in a particular product or service being offered by a vendor, such as an automobile or travel agency services. Accordingly, at step 110, a transaction data picture is requested from the user of such program. This operation takes place, for example, after preliminary data concerning the transaction is first obtained, such as date of the purchase, number of shares, price per share and the like. After this, the user is presented at step 115a with a visual window identifying both a pallette of parameters (representing assertions, reasons, motivations, etc.) and at 115b with a data canvas for creating a customized data picture depicting the user's total rationale for the transaction in question. It should be noted that for ease of use, and as shown in FIG.2 , the palette and canvas are presented visually to the user at the same time. As will be apparent from the description below, the palette and canvas present a simple, single data collection screen for the user to capture all of the information associated with a particular action/transaction during a particular session. The electronic windows for such palette and canvas are generated using well-known programming techniques, and the specifics of the same are not material to the present invention except as described herein. Nonetheless, a listing of the important source code routines used in the present invention is appended to the end of the present disclosure as Appendix A.

During an interactive session at step 120 (which can be thought of as a data "painting" session) the user is permitted to select from the list of parameters (which again can be reasons, assertions, etc.) presented, and to drag and drop any particular item as desired to a location on the data canvas. In this manner, the user can quickly, easily and intuitively manipulate subjective data parameters into a more concrete and structured format, representing a data "picture" without having to plow through numerous irrelevant data queries, and without having to maintain a precise mental

arithmetic valuation (or relative ranking) of the data parameters selected. Instead, the vertical (and horizontal) placement of such data parameters as arranged within such window by the user can be used as ranking indicators. At the end of the data "painting" session at step 125 the data picture is converted into a series of data values in a fashion completely transparent to the user, and without any additional effort on their behalf. These data values are in turn either stored or passed on to an application program at step 130. The interface then yields control back to the main application program at 135.

FIG. 2 illustrates one embodiment of the inventive interface 200 as seen by a user at step 120. On the left hand side of the window, a menu list 205 of parameters 221, 222, etc. are identified. This set is presented visually to the user in what is known in conventional terms as a tree-structured menu. Preferably, for ease of use, this tree menu 205 has a shallow depth (i.e., number of levels) and reasonable breadth (i.e., number of items per level) that is manageable, and which permits a user to visualize all possible selections without having to perform time consuming scrolling operations. Generation of tree-structured menus is well-known in the art, and so will not be discussed at length here. The use of a tree-structured menu 205 is especially advantageous within the context of the present invention, nonetheless, since it permits a user to rapidly identify those reasons affecting or motivating their behavior vis-a-vis the transaction in question. Again, in the present preferred embodiment, the reasons and assertions presented in menu 205 are those factors commonly associated with actions or transactions (buying or selling) securities or options. These include such factors as technical charting indicators, specific events that may affect the stock price (a split announcement for example), rumors, tips from friends, etc. These factors are broken into larger category groups at a first level 220, and in more fine detail at a sub-category second level 221. For instance, category group "Chart Pattern" 220 is broken down into 6 sub-categories 221 identified as "Inverted Head & Shoulders," "Cup with Handle," "Broke above Trendline," Broke below Trendline," "Broke below support," and "Rounded Top," all of which are commonly understood technical analysis tools in the field of financial instrument trading. It should be apparent that such category and sub-categories are merely illustrative of the parameters that could be identified in

interface 200, and that the present invention is not limited to any specific set of the same. For example, in an interface geared towards understanding the motivations of a prospective consumer in the automotive field, category sets may include quality, warranty, price, performance, luxury, reliability, and so on. Sub-categories for price might be broken down into numerical sets ranging from 10k to 15k, 15k to 20k, 20-25k, 25-30k, etc. The particular implementation of the parameter set, of course, can be tailored to the features/functions associated with the item in question, so that reasonable allowance is made for variations in the characteristics of such item.

Another advantage of the present invention lies in the fact that the menu 205 can be supplemented by the user to include new categories 221 reflecting particular assertions, reasons, motivations, influences, etc., experienced by such user. As an example, a new category "other" 221 can be added to menu 205 to include such sub-categories as "Read in local newspaper," "Heard good things from John X," etc. Thus menu 205 can be customized in part, and this provides additional flexibility to accommodate particular influences in the user's realm of experience. Such additions to menu 205 can be effectuated using any number of known techniques in the art.

Accordingly, interface 200 presents the user with a complete and customized set 205 of parameters that are applicable to the transactions handled by an underlying application program. Moreover, such parameters are depicted in a manner that avoids prior art cumbersome multiple-page formats, which tend to confuse and slow down interaction between the user and the application program. The arrangement of menu 205 in fact, presents the user essentially with an electronic data pallette in a first portion of the interface, which he/she can easily visualize in complete form at a glance and utilize to create a data picture 210 on a data canvas 215 (or preference field) in a second portion of the interface as shown on the right hand side of interface 200.

In the example given in FIG.2, data picture 210 includes two subjective parameters 230, 231 selected by a user from menu 205 as motivating his/her to purchase a particular stock. In this instance, such user has identified certain assertions as the basis for his/her action, including the fact that they are influenced by their perception that the company has a "Good Dividend," and that the "Company has good products." The selection of parameters 230 and placement in preference field/data

canvas 215 is done with a conventional computer mouse in a drag and drop manner well-known in the art, and as illustrated generally in FIG.2 by the path of the dotted lines shown there. Other techniques, such as direct control devices (light pens, touch-screens, and so on) or indirect control devices (touch pads, joysticks, trackballs etc.), can also be used in lieu of a mouse for the painting of data picture 210. The paramount consideration, in this respect, is that an easily manipulable tool should be selected to permit the user to interact quickly and intuitively with palette 220 and move objects (data parameters) to data canvas 215 with ease.

In this preferred embodiment, preference field/data canvas 215 is presented in a relevance spectrum format, with descriptors at the top/bottom providing visual feedback to the user to identify a value of importance for the identified parameter, ranging from "Less" to "More" important. This allows the user to not only identify the data parameters associated by them in connection with the transaction, but also to rank such parameter in a more objective fashion, both absolutely in terms of a perceived importance, but also relative to other factors. This ranking, too, can be done quickly by the user since it is done without the need for cumbersome arithmetic computations and/or manipulation of electronic sliding scales, etc. In fact, the only physical operations required of the user are the selection (preferably using a common computer mouse) of parameters on the left hand side of the interface, and their drag and drop placement on the right hand side of the interface. As the pallette 205 and canvas 215 are located closely and adjacent to each other spatially, this task can be performed extremely ergonomically and rapidly by a user utilizing a computer mouse. In most cases, in fact, the distance required for the user to move a data parameter from palette 205 to canvas 215 is relatively small: i.e., less than half the total interface screen size. This reduction in demands on the user's attention and time results in more thorough and accurate data extraction because the user is not discouraged or de-motivated by the interface from presenting his/her opinions in complete form, and in a manner that is more intuitive and in tune with the way in which most human operators work most comfortably - namely, with a "picture" of their impressions, rather than in hard numerical representations. Moreover, after such data is captured, it can be retrieved and modified at will by the user, permitting them to perfect and learn from their

mental impressions of actual or proposed actions/transactions.

These advantages of the present invention can be illustrated very easily with simple example. In a typical prior art program interface, the user is required to parse through page after page of questions, assigning numerical values to each parameter of interest. This is slow because the user is confronted with an ocean of choices that may be of no importance, but which yet must still be navigated to satisfy the data collection goals of the underlying application. In the present invention, the user need only utilize those data parameters of interest to them from a data parameter pallette. Then, as the user traverses a typical prior art interface, they may assign a number of relevance values R1, R2...Rk, and the like for each identified parameter. A substantial mental burden is imposed on the user in both remembering the relevance values previously provided for prior parameters, and in attempting to effect a relative ranking of new parameters within the framework of the data collection effort at that point. In other words, the user may associate two parameters with the same relevance, but if information on the two parameters is solicited at different times during the data collection interview, he/she is likely to forget the value previously assigned. This means that the data value given may be inaccurate, and the data collection effort is degraded. Furthermore, when confronted with the fact that he/she has previously identified two parameters as having a relative ranking of say 6 and 9, a third parameter lying between these two in importance must be assigned an arithmetic value that is calculated mentally by the user. A value for a fourth parameter, perhaps associated by the user between the third and second parameter in importance, must then be calculated by the user. It is apparent that this mental effort is both time consuming and potentially frustrating. In contrast, in the present invention, users can easily and dynamically arrange and re-arrange their identified priorities through simple physical manipulation, and without the need for tagging these parameters with hard, fixed arithmetic values. Instead, the underlying application program can simply process the data picture provided, and then carry out a computational evaluation of the necessary values to be associated with the data items arranged in the data picture. This fact, too, reduces significantly the time associated with a data interview, and is likely to result in greater user appreciation for (and utilization of) the underlying

application programs.

A further advantage provided by the present invention lies in the fact that since no keyboard typing entries are required of the user, and no electronic "check boxes" or the like are filled in, errors in the data capture process are also reduced, thus further increasing the reliability of data captured using the inventive interface. In fact, the possibility of errors is minimized in large part because the user is permitted to see the entirety of his/her data entry visually for sufficient periods of time to allow for detection of obvious errors. This and other aspects of the present invention therefore make optimal use of visual information processing centers available to human operators.

Finally, a useful industry accepted predictive model of hand movement complexity in an interface (i.e., moving a pointing device to one region of a display to another) is provided by a formula in Schneiderman, at page 325. This formula states that:

$$\text{Index of difficulty} = \text{Log}_2(2D/W).$$

In this formula, D = distance that must be traveled across the screen by a pointing device (such as a mouse), and W is the area of the target for the pointing device. From this simple model, it is readily apparent that the interface proposed by the applicant is optimized to reduce user interaction difficulties. This is because, as seen in FIG.2, the travel distance (D) for the user's mouse in moving data parameters from the menu to data canvas 215 is kept very small; conversely, the area (W) for data canvas 215 is relatively large. All things being equal, this mathematical model illustrates why an index of difficulty for the present invention is relatively small as compared to prior art interfaces which rely on the operator's *ability to move* a cursor across an entire screen (to address an item of interest) and then require fairly accurate control to land within a small designated area for expressing some value for such item of interest. A time for an operator to achieve such movement, of course, is directly related to such index as indicated by Schneiderman; again, for the present invention, this predictive modeling helps explain why an interface of the type described herein is much faster for even a novice operator, and thus more likely to be endorsed by the same.

Similarly, at pp. 391 - 397, Schneiderman addresses the issue of display layout "appropriateness," as measured by a series of metrics discussed therein, including task frequency and sequencing characteristics. An evaluation of the layout of the present invention by such metrics reveals, quite distinctly, that an optimal arrangement is provided from the user's perspective, because task sequences are kept extremely simple (the user need only perform one physical operation to input and classify a data parameter), and the physical arrangement of the interface (relative placement of data menu and data canvas) is implemented such that such physical operation is effectuated by a single motion that occurs over a very short distance, and with high accuracy. This minimizes the amount of visual scanning (and thus time and burden) required by the user to express his/her thoughts, because an interaction area is minimized as well. Since related ideas are grouped in physically contiguous areas in menu 205, this further reduces eye strain, mental strain, etc.

To further enhance the appearance and utility of data canvas 215, color, shading, highlighting and other visual indicators can be used to provide feedback to the user as well. In a preferred embodiment data canvas 215 includes a blue background, and is arranged so that a "gradient" is presented to the user in the form of shading arranged from darkest (more important) to lightest (least important). This arrangement is chosen to provide the user with a pleasant visual environment, and as a visible metaphor/reinforcer for the fact that factors weighing "heavier" in his/her mind should be placed in a heavier shade portion of the gradient. In addition, as individual data parameters 230 are moved on data canvas 215, nearby already-placed data parameters (or the appropriate spectrum legend at one end of the gradient), can "glow" or become highlighted to remind the user that they are moving the selected data parameter to a region of relevance that is close to the previously identified data parameter. This feature, too, helps the user to orient and rank his/her reasons, preferences, opinions, etc. in a more orderly and reliable fashion, and without the need for arithmetic computations. Another visual enhancement that could be made, of course, is the addition of scaling information - for instance, some form of ruler markings as used in conventional word processing layout programs, or drafting programs - along the edges of canvas 215. Such markings could include numerical

indicators ranging, for example, from 1 to 10, or other convenient divisions to better identify the gradient. Other variations are of course possible, and will apparent to those skilled in the art based on studies of human perceptual skills and traits and from the teachings herein.

When the user has completed the creation of data picture 210, it can then be saved and stored using conventional techniques as a transaction "reasons" file (or entry) for later use by an underlying application program. A conversion takes place (see FIG.1, step 125) so that the user data picture can be utilized by an underlying application program. The specifics of such conversion will of course vary from application to application, depending on the nature of the specific data parameters presented to the user, the nature of the specific data canvas, etc, etc. In the present embodiment, an identification is first made of all the data parameters 215 making up data picture 210 on data canvas 215. The placement of such data parameter within data canvas 210 is also noted; in this instance, only the vertical distance component is retained, but it should be apparent that a horizontal component could also be used.

Thus, a typical data picture record can include a transaction identifier, the identity of a data parameter, and a location placement within the gradient (in arbitrary units). Other information can of course be included in such record and many different variations are possible for such data picture record from application to application. Each data picture 210, therefore, will consist of one or more of such data picture records, thus constituting a complete transaction record reflecting a complete capture of the user's perceptions, motivations, reasoning, etc., for a particular transaction. At this point, as indicated at step 130 (FIG. 1) transaction record, and all of its underlying components are available for use by an applications program as needed. In a preferred embodiment, such transaction records are maintained and retrievable at later times by an operator so that personalized lessons can be learned from correlations of the user's data pictures (describing the user's particular rationale for a transaction) and resulting gain or loss from such particular stock purchase or sale transactions.

It should be noted that in addition to the vertical placement component value retained, a horizontal placement component might be used, for example, where data canvas 210 also has a horizontal spectrum (left to right orientation) depicting a

different consideration, i.e., such as the user's perception of the credibility of such identified parameter. As an example, a user might identify a rumor as strongly motivating his/her behavior (thus placing the data parameter high on the data canvas), but could also qualify this belief by indicating that the perceived truthfulness of the rumor is low by placing the data parameter on the far left (or right) according to an "accuracy" spectrum. This feature permits the present interface to better capture the "fuzzy" logic of human actions and perceptions through an additional variable qualifying the data parameters identified for a particular transaction.

For purposes of the present invention a detailed description of those features commonly found and understood in application windowing technology (i.e., such as sizing, scrolling, handling and the like) is not provided. Such features can be implemented in any one of many techniques known in the art, and the invention is not limited in any way by such specific implementations.

Consequently, the present invention permits a user of an application program to enter data parameters in an efficient, intuitive, rapid fashion using a single data collection window which captures all of the subjective information in a single snap shot. And, through the manipulation of only a few visual objects representing individual perceptions, motivations, reasons, etc., an underlying application program can capture more relevant data more efficiently than through the use of prior art interfaces.

In addition, the present invention allows, for the first time, for a user to convey his/her mental impressions in a form understood by him/her, and yet parseable in an intelligent fashion by an underlying program. The present inventive interface, in fact, can serve as a simple, non-intimidating environment and mechanism to permit even novice users to interact with sophisticated and complex application programs that would otherwise be avoided. This interaction is expected to increase as application programs begin to make more and more use of "fuzzy" data passed on by users of the same.

In another variation of the present invention, the underlying application program can track prior transaction reasons records created by the user. From this tracking, an evaluation can be made of the N most common data parameters identified

by the user, as well as their average placement on data canvas 215. This information, in turn can be used to generate an "expected" data picture 210,' which, when a new transaction reasons record is to be generated (i.e., at step 110) can be presented to the user (at his/her option) as a template to work from. At that point, the user can accept, modify, supplement such data picture 210' based on the particular details of the transaction in question. In this manner, the burden on a user to input relevant data is further minimized.

For added functionality, the underlying application program can also dynamically process data picture 210 into a transaction record, and then visually display a correlation of such transaction record with other transaction records 210' previously entered by the user. In this manner a user can quickly and visually identify similarities, trends, etc. in his/her rationale (or other people's rationales) for such transactions. This feedback window 260 can be generated using any one of conventional programming techniques based on the nature of the underlying data, the perspective desired by the user, the desired sub-set of prior transaction records, the desired correlation, etc., and can be presented alongside graphical interface 200 as illustrated generally in FIG. 2. As a simple example, in the case of a financial trading context, the user could request a comparison chart in feedback window 260 illustrating the overall financial gain/loss in a graph 261 incurred by such user (based on evaluation of prior transaction records) when they (or other users) had identified "Good Dividend," and the "Company has good products" (or some other reasons) as reasons for their purchasing a particular financial instrument. The overall average could be plotted as well as a separate graph 262 as noted. Other variations of feedback window 260 and information to be displayed thereon are of course possible, and the present invention is not limited to any particular implementation. For example, instead of a chart, a simple icon resembling a human face smiling or frowning can communicate positive or negative feedback information, letting the user know immediately whether they are getting closer or further away from an ideal data picture. This aspect of the present invention allows the user to immediately visualize the expected results of the action/transaction based on that person's unique experiences/rationales. This feature is extremely beneficial as it can serve to prevent

obvious errors, or to suggest a course of action that is likely to be favorable based on an evaluation of prior transaction data. Other potential applications which can utilize this kind of immediate visual feedback including telemarketing, product service support, etc. In such environments, the present interface could be used as a marketing data capture interface, and as data is collected by an operator from a customer, feedback window 260 can present potential options to the operator correlating such user's data with prior data transactions, or against some set of criteria. As one example, in response to various customer descriptions of problems with a product, an operator could be given a visual or symbolic list of potential areas that are the origin of the same. This is but one example, of course, and the present invention can be advantageously used in any number of similar environments.

In yet another variation, data parameters 220 can of course be presented in symbolic, or iconic form, rather than as descriptive text objects. For example, in an application program soliciting feedback on travel modes used by a prospective traveler, menu 220 may instead contain a series of graphical icons representing conventionally accepted depictions of trains, planes, cars, taxis, bicycles, cruise ships, etc. In an application program for capturing entertainment interests, iconic representations of movies, Cds, LP records, television, etc. could be presented as data parameters. Other examples will be apparent to those skilled in the art for other fields of interest.

In another embodiment of the inventive interface 200, instead of corresponding to "reasons" employed by a user for a particular action, parameters 221, 222, etc. in menu 205 correspond generally to "lessons" learned from a particular action/transaction. This way, during an interactive session at step 120 the user is permitted to paint a data picture explaining what he/she learned from the particular experience of engaging in the action/transaction. Preferably this second variation of the inventive interface is used in conjunction with, and as a complement to the first embodiment noted above. For example, after first creating a data picture with the aforementioned graphical interface to identify particular reasons and motivations for a particular action/transaction, the user can then invoke a separate interface at a later time for identifying any subsequent lessons, understandings, etc. that he/she has

acquired or associates with the particular action/transaction. This session then results in a second data picture 210 associated with lessons specific to such user which they can store, modify, and learn from to improve their objectivity when participating in such actions/transactions. As an example, in the case of a financial trading context, parameters 205 can list such options as "sold too early," "sold too late," "got in too early," "got in too late," "don't listen to advice from X," etc. This information would be captured by the interface so the user can maintain a permanent but modifiable diary or log of useful data which can be referred to at a later time to evaluate their performance, to see what weaknesses or strengths they exhibit, to see what events seem to influence their thinking, and to observe trends in the same.

As is apparent, for ease of use for the user, this second interface only varies from the first interface described above in the identity of the parameters provided in menu 205, which, again, need only identify particular "lessons" instead of "reasons" in this case. If desired, data canvas 215 can also be modified nonetheless with suitable language adjustments to the spectrum legend to correlate better with the parameters provided in menu 205. Again, as with the first interface, the user can rank the relative perceived importance of the lessons learned, which acts as strong visual feedback when reviewed at a later time.

To better capture the manner in which individuals collect information, the data pictures 210 created by the second interface can also be modified at a later time to reflect new insights garnered by the user. For instance, after closing out a losing transaction involving an equity purchase and sale, the user may identify the fact that they bought too early as a lesson that they learned. At a later time, in the event the equity price rebounds, they may also conclude that they sold too early as well. Because the data picture is stored electronically, it can be retrieved via interface 200 and modified at will by the user to reflect such new understandings and lessons about the action/transaction. This functionality permits the user to better grasp his/her strengths and weaknesses at a glance, and helps reinforce the lessons learned from such actions/transactions.

While the present invention has been described in terms of a preferred embodiment, it will be apparent to those skilled in the art that many alterations and

modifications may be made to such embodiments without departing from the teachings of the present invention. For example, it is apparent that the present invention would be beneficial used in any applications program where it is desirable to obtain accurate, reliable information from a user in an efficient, intuitive fashion. Other types of particular implementations for the data parameter menu and data canvas beyond those illustrated in the foregoing detailed description can be used suitably with the present invention. Accordingly, it is intended that the all such alterations and modifications be included within the scope and spirit of the invention as defined by the following claims.

APPENDIX A

```
Content-Type: text/java;
        name="ReasonDragPanel.java"
Content-Transfer-Encoding: quoted-printable
Content-Disposition: attachment;
        filename="ReasonDragPanel.java"
import java.awt.*;
import java.util.Vector;
public class ReasonDragPanel          extends Panel
{
        protected Image im1;    // offscreen image
        protected Graphics g1 =3D null;  // offscreen graphics context
        private int viewHeight =3D 300;
        private int viewWidth  =3D 300; // pixel size of tree display
        private Color bgHighlightColor =3D Color.blue:    // selection bg color
        private Color fgHighlightColor =3D Color.white;   // selection fg color
        private Color fgColor =3D Color.black;
        private Color bgColor =3D Color.white;
        private Vector drags;
        private FontMetrics fm; // current font metrics
        private int mouseX =3D -1;
        private int mouseY =3D -1;
        private int grabX =3D 0;
        private int grabY =3D 0;
        private DragTreeNode selectedNode =3D null;
        private boolean dragging =3D false;
        private boolean dragUp =3D false;
        private boolean dragDown =3D false;
        private Color topColor =3D new Color(128, 128, 255);
        private Color bottomColor =3D new Color(255, 255, 255);
        private int fadeSteps =3D 20;
```

```
        public String topLabel =3D "More Important";
        public String bottomLabel =3D "Less Important";
=09
        public ReasonDragPanel()
        {
                drags =3D new Vector(5, 1);
        }
        public Vector getDrags()
        {
                return drags;
        }
        public void addNode(Reason r, int y)
        {
                for (int i =3D 0; i < drags.size(); i++)
                        if
(((DragTreeNode)drags.elementAt(i)).reason.name.equals(r.name))
                                return;
                for (int i =3D 0; i < drags.size(); i++)
                        if (!((DragTreeNode)drags.elementAt(i)).dragged)
                                drags.removeElementAt(i--);
                drags.addElement(new DragTreeNode(r, 10, y, !(y =3D=3D 10)));
                repaint();
        }
        Color fadeColors(Color f, Color b, double per)
        {
                double nper =3D 1 - per;
                int r1 =3D (int)(f.getRed() * per + b.getRed() * nper);
                int g1 =3D (int)(f.getGreen() * per + b.getGreen() * nper);
                int b1 =3D (int)(f.getBlue() * per + b.getBlue() * nper);
                return new Color(r1, g1, b1);
        }
```

```
public void update (Graphics g)
{
        redraw();
   paint(g);
}
=09
public void paint (Graphics g)
{
        Dimension d =3D size();
=09
        if ((d.width !=3D viewWidth) || (d.height !=3D viewHeight))
                redraw();
        g.drawImage(im1, 0, 0, this);
}
public void redraw()
{
        drawDrag();
}
public void drawDrag()
{
        Dimension d =3D size();
=09
        if ((d.width !=3D viewWidth) || (d.height !=3D viewHeight) || =
(g1=3D=3Dnull))
        {
                im1 =3D createImage(d.width, d.height);
                if (g1 !=3D null) {
                        g1.dispose();
                }
      g1 =3D im1.getGraphics();
      viewWidth=3Dd.width;
```

```
                viewHeight=3Dd.height;
                }
        =09
                Font f =3D null;//getFont();  //unix version might not provide a = default
        font
                if (f =3D=3D null)
                {
                f =3D new Font("Helvetica", Font.PLAIN, 13);
                g1.setFont(f);
                setFont(f);
                }
                    if (f !=3D null)
                    {
                            if (g1.getFont() =3D=3D null)
                                    g1.setFont(f);
                    }
                =09
                fm =3D g1.getFontMetrics();
                g1.setColor(bgColor);
                g1.fillRect(0,0,viewWidth,viewHeight);  // clear image
                    /*for (int i =3D 0; i < fadeSteps; i++)
                    {
                            g1.setColor(fadeColors(bottomColor, topColor, 1.0*
        i/fadeSteps));
                            g1.fillRect(0, size().height * i/fadeSteps, size().width, =
        1+size().height / fadeSteps);
                    }*/
        =09
        =09
                    if (!dragging)
                            selectedNode =3D null;
```

```
                    for (int i =3D 0; i < drags.size(); i++)
                            drawNode((DragTreeNode)drags.elementAt(i));
                    /*g1.setColor(Color.red);
                    for (int i =3D 0; i <=3D 100; i+=3D10)
                            g1.drawString(""+i, 20, valueToY(i));*/
                    //if (dragging)
                    {
                  f =3D new Font("Helvetica", Font.BOLD + Font.PLAIN, 14);
          g1.setFont(f);
                            if (dragUp)
                                    g1.setColor(Color.red);
                            else
                                    g1.setColor(fgColor);
                            g1.drawString(topLabel. (size().width -
          fm.stringWidth(topLabel)) >> =
          1, fm.getHeight() + 2);
                            if (dragDown)
                                    g1.setColor(Color.red);
                            else
                                    g1.setColor(fgColor);
                            g1.drawString(bottomLabel, (size().width - =
          fm.stringWidth(bottomLabel)) >> 1, size().height - fm.getHeight() + 2);
                    }
                    //g1.setColor(fgColor);
                    //g1.drawRect(0,0,viewWidth, viewHeight - 1);
            }
            private int topLine()
            {
                    Font f2 =3D g1.getFont();
              Font f =3D new Font("Helvetica", Font.BOLD + Font.PLAIN, 14);
              g1.setFont(f);
```

```
                int retVal =3D fm.getHeight() + 5;
                g1.setFont(f2);
                return retVal;
        }
        private int dragHeight()
        {
                return size().height - (3*topLine() + 4);
        }
        private int valueToY(int val)
        {
                return topLine() + 4 + (dragHeight() * val / 100);
        }
=09
        private void drawNode(DragTreeNode node)
        {
                int width =3D fm.stringWidth(node.reason.name);
                int height =3D fm.getHeight() + 4;
                boolean mouseIn =3D false;
                int realy =3D valueToY(node.y);
                int realx =3D node.x;
        =09
                if (dragging && (selectedNode =3D=3D node))
                {
                        realx -=3D grabX;
                        realy -=3D grabY;
                }
                g1.setColor(fgColor);
                if (((mouseX >=3D realx) && (mouseY >=3D realy) &&=20
                        (mouseX < realx + width+10) && (mouseY < realy +
height+4)) ||=20
                        (dragging && (selectedNode =3D=3D node)))
```

```
                {
                        mouseIn =3D true;
                        if (!dragging)
                        {
                                selectedNode =3D node;
                                grabX =3D mouseX - realx;
                                grabY =3D mouseY - realy;
                        }
                }
                if (mouseIn)
                        g1.setColor(bgHighlightColor);
                else
                        g1.setColor(bgColor);
                g1.fillRect(realx, realy, width + 10, fm.getHeight() + 4);
                if (mouseIn)
                        g1.setColor(fgHighlightColor);
                else
                        g1.setColor(fgColor);
                g1.drawRect(realx, realy, width + 10, fm.getHeight() + 4);
                g1.drawString(node.reason.name, realx+5, realy+fm.getHeight());
        /*      if (dragging && (selectedNode =3D=3D node))
                {
                        g1.setColor(Color.red);
                        g1.drawString(""+node.y, realx, realy+2+(fm.getHeight() * 2));
                }*/
        }
=09
=09
=09
        public synchronized Dimension preferredSize()
        {
```

```
    return new Dimension(175, 125);
  }
=09
  public synchronized Dimension minimumSize()
  {
    return new Dimension(50, 50);
  }
  public boolean mouseMove(Event e, int x, int y)
  {
      mouseX =3D x;
      mouseY =3D y;
      repaint();
      return true;
  }
  public boolean mouseDrag(Event e, int x, int y)
  {
      dragUp =3D (y < mouseY);
      dragDown =3D (y > mouseY);
      if (selectedNode !=3D null)
      {
          dragging =3D true;
          selectedNode.dragged =3D true:
          selectedNode.x =3D x;
          selectedNode.y =3D 100 * (y-topLine()) / dragHeight();
      }
      repaint();
      return true;
  }
  public boolean mouseUp(Event event, int x, int y)
  {
      if (dragging)
```

```
                {
                        selectedNode.x -=3D grabX;
                        selectedNode.y -=3D (100 * grabY / dragHeight());
                        if (selectedNode.y < 0)
                                selectedNode.y =3D 0;
                        if (selectedNode.y > 100)
                                selectedNode.y =3D 94;
                        if ((selectedNode.x < 0) && (selectedNode.x + =
fm.stringWidth(selectedNode.reason.name) + 10) > 0)
                                selectedNode.x =3D 0;
                        if ((selectedNode.x < size().width) && (selectedNode.x + =
fm.stringWidth(selectedNode.reason.name) + 10 > size().width))
                                selectedNode.x =3D size().width - =
(fm.stringWidth(selectedNode.reason.name) + 10);
                        Rectangle r =3D new Rectangle(size());
                        Rectangle r2 =3D new Rectangle(selectedNode.x,
selectedNode.y,
                                        fm.stringWidth(selectedNode.reason.name) +
10, fm.getHeight()+4);
                        if (!r.intersects(r2))
                                deleteNode(selectedNode.reason.id);
                        selectedNode =3D null:
                        dragging =3D false;
                        dragUp =3D false;
                        dragDown =3D false;
                        repaint();
                }
          return true;
        }
        public void deleteNode(int id)
        {
```

```
                    for (int i =3D 0; i < drags.size(); i++)
                    {
                        if (((DragTreeNode)drags.elementAt(i)).reason.id =3D=3D id)
                            drags.removeElementAt(i);
                    }
                }
            }
Content-Type: text/java;
        name="ReasonTree.java"
Content-Transfer-Encoding: quoted-printable
Content-Disposition: attachment;
        filename="ReasonTree.java"
import java.awt.*;
import java.util.Vector;
import TreeNode;
public class ReasonTree       extends Panel
{
        public static final int CHILD   =3D 0;
        public static final int NEXT    =3D CHILD + 1;
        public static final int LAST    =3D CHILD + 2;
        public static final int SEL_CHANGED =3D 1006; //selection changed event
=09
        private TreeNode rootNode;          // root node of tree
        private TreeNode selectedNode;      // highlighted node
        private TreeNode topVisibleNode;    // first node in window
=09
        Scrollbar  sbV;                         // vertical scrollbar
        int                 sbVPosition=3D0;    // hold value of vertical scrollbar
        int                 sbVWidth;       // width of vertical scrollbar
        long            sbVTimer =3D -1;     // time of last vert scrollbar event
        private int count=3D0;   // Number of nodes in the tree
```

```
        private int viewCount=3D0;// Number of viewable nodes in the tree
        private Color bgHighlightColor =3D Color.blue;   // selection bg color
        private Color fgHighlightColor =3D Color.white;  // selection fg color
        private int viewHeight =3D 300;
        private int viewWidth  =3D 300; // pixel size of tree display
        private int viewWidest =3D 0 ;  // widest item displayable (for horz =
scroll)
        int cellSize   =3D 16;  // size of node image
        int clickSize  =3D 8;   // size of mouse toggle (plus or minus)
        int imageInset =3D 3;   // left margin of node image
        int textInset  =3D 6;   // left margin for text
        int textBaseLine=3D 3;   // position of font baseline from bottom of =
cell
        private FontMetrics fm; // current font metrics
        long timeMouseDown;    // save time of last mouse down (for double =
click)
        int doubleClickResolution =3D 333; // double-click speed in =
milliseconds
        Portfolio applet;
        protected Image im1;   // offscreen image
        protected Graphics g1 =3D null;  // offscreen graphics context
        int prevX =3D -1;
        int prevY =3D -1;
        private String descrip =3D null;
=09
        public ReasonTree(Portfolio a)
        {
          super.setLayout(new BorderLayout());
          add("East", sbV =3D new Scrollbar(Scrollbar.VERTICAL));
               applet =3D a;
        }
```

```
public void clearTree()
{
        rootNode =3D null;
        selectedNode =3D null;
        topVisibleNode =3D null;
}
public void setBackground(Color c)
{
   super.setBackground(c);
   repaint();
}

public Color getBackground()
{
   return super.getBackground();
}
=09
=09
public void setForeground(Color c)
{
   super.setForeground(c);
   repaint();
}
=09
=09
public Color getForeground()
{
   return super.getForeground();
}
=09
=09
```

```
public void setFgHilite(Color c)
{
  fgHighlightColor =3D c;
  repaint();
}
=09
/**
 * Gets the current foreground selection hilite color.
 * @return the current foreground selection hilite color
 * @see #setFgHilite
 */
public Color getFgHilite()
{
  return fgHighlightColor;
}
=09
/**
 * Gets the current background selection hilite color.
 * @return the current background selection hilite color
 * @see #getBgHilite
 */
public void setBgHilite(Color c)
{
  bgHighlightColor =3D c;
  repaint();
}
=09
/**
 * Gets the current background selection hilite color.
 * @return the current background selection hilite color
 * @see #setBgHilite
```

```
        */
        public Color getBgHilite()
        {
          return bgHighlightColor;
        }
=09
        public void insert(TreeNode newNode, TreeNode relativeNode, int =
position)
        {
          if (newNode=3D=3Dnull || relativeNode=3D=3Dnull)
            return;
          =20
          if (exists(relativeNode)=3D=3Dfalse)
            return;
          =20
          switch (position)
          {
            case CHILD:
              addChild(newNode, relativeNode);
              break;
              =09
            case NEXT:
              addSibling(newNode, relativeNode, false);
              break;
              =09
            case LAST:
              addSibling(newNode, relativeNode, true);
              break;
=09
            default:
              // invalid position
```

```
                return;
              }
            }
=09
  /**
   * Returns the "root" node. The root node is the top-most node
   * in the tree hierarchy. All other nodes stem from that one.
   * @return the root tree node
   */
      public TreeNode getRootNode()
      {
        return rootNode;
      }
=09
      /**
       * Returns the total number of nodes in the tree.
       */
      public int getCount()
      {
        return count;
      }
=09
      /**
       * Returns the total number of viewable nodes in the tree.
   * A node is viewable if all of its parents are expanded.
   */
      public int getViewCount()
      {
        return viewCount;
      }
=09
```

```
        /**
         * Determines if the given node is viewable.
      * A node is viewable if all of its parents are expanded.
      * @param node the node to check
      * @return true if the node is visible, false if it is not
      * @see #viewable(java.lang.String)
         */
        boolean viewable(TreeNode node)
        {
          for (int i=3D0; i<viewCount; i++)
          {
            if (node =3D=3D v.elementAt(i))
            {
              return true;
            }
          }
=09
          return false;
        }
=09
        /**
         * Determines if the node with the given text is viewable.
      * A node is viewable if all of its parents are expanded.
      * @param s the node text to find
      * @return true if the node is visible, false if it is not
      * @see #viewable(TreeNode)
         */
        boolean viewable(String s)
        {
          if (s=3D=3Dnull)
          {
```

```
            return false;
          }
=09
          for (int i=3D0; i<viewCount; i++)
          {
            TreeNode tn =3D (TreeNode)v.elementAt(i);
=09
            if (tn.getText() !=3D null)
            {
              if (s.equals(tn.getText()))
              {
                return true;
              }
            }
          }
=09
          return false;
        }
=09
        /**
         * Determines if the given node is in the ReasonTree.
       * @param node the node to check
       * @return true if the node is in the ReasonTree, false if it is not
       * @see #exists(java.lang.String)
         */
        public boolean exists(TreeNode node)
        {
          recount();
=09
          for (int i=3D0; i<count; i++)
          {
```

```
                if (node =3D=3D e.elementAt(i))
                {
                  return true;
                }
              }
=09
              return false;
            }
=09
            /**
             * Determines if the node with the given text is in the ReasonTree.
             * @param s the node text to find
           * @return true if the node is in the ReasonTree, false if it is not
           * @see #exists(TreeNode)
             */
            public boolean exists(String s)
            {
              recount();
=09
              if (s=3D=3Dnull)
              {
                return false;
              }
=09
              for (int i=3D0; i<count; i++)
              {
                TreeNode tn =3D (TreeNode)e.elementAt(i);
=09
                if (tn.getText() !=3D null)
                {
                  if (s.equals(tn.getText()))
```

```
                {
                  return true;
                }
              }
            }
=09
            return false;
          }
=09
          // add new node to level 0
          /**
           * Adds a new node at root level. If there is no root node, the given
           * node is made the root node. If there is a root node, the given node
           * is made a sibling of the root node.
           * @param newNode the new node to add
           * @see #insert
           */
          public void append(TreeNode newNode)
          {
            if (rootNode=3D=3Dnull)
            {
              rootNode=3DnewNode;
              selectedNode =3D rootNode;
              count=3D1;
            }
            else
            {
              addSibling(newNode, rootNode, true);
            }
          }
=09
```

```
void addChild(TreeNode newNode, TreeNode relativeNode)
{
  if (relativeNode.child =3D=3D null)
  {
    relativeNode.child =3D newNode;
    newNode.parent =3D relativeNode:
    count++;
  }
  else
  {
    addSibling(newNode, relativeNode.child, true);
  }
=09
  relativeNode.numberOfChildren++;
}
=09
void addSibling(TreeNode newNode, TreeNode siblingNode)
{
    addSibling(newNode,siblingNode,true):
}
=09
void addSibling(TreeNode newNode, TreeNode siblingNode, boolean =
asLastSibling)
{
    if (asLastSibling)
    {
        //Find last sibling
        TreeNode tempNode =3D siblingNode;
        while (tempNode.sibling !=3D null)
            tempNode =3D tempNode.sibling;
    =09
```

```
                    tempNode.sibling =3D newNode;
                }
                else
                {
                    //Insert the newNode below the siblingNode
                    newNode.sibling =3D siblingNode.sibling;
                =09
                    siblingNode.sibling =3D newNode;
                }
            =09
                //Set the parent of the new node to the parent of the sibling
                newNode.parent =3D siblingNode.parent;
            =09
                count++;
            }
        =09
        =09
            private Vector e; // e is vector of existing nodes
            private void recount()
            {
              count =3D 0;
              e =3D new Vector();
        =09
              if (rootNode !=3D null)
              {
                rootNode.depth=3D0;
                traverse(rootNode);
              }
            }
        =09
            private void traverse(TreeNode node)
```

```
{
  count++:
  e.addElement(node);
=09
  if (node.child !=3D null)
  {
    node.child.depth =3D node.depth+1;
    traverse(node.child);
  }
  if (node.sibling !=3D null)
  {
    node.sibling.depth =3D node.depth;
    traverse(node.sibling);
  }
}
=09
private Vector v; // v is vector of viewable nodes
private void resetVector()
{
    // Traverses tree to put nodes into vector v
    // for internal processing. Depths of nodes are set,
    // and viewCount and viewWidest is set.
    v =3D new Vector(count);
    viewWidest=3D30;
=09
    if (count < 1)
    {
        viewCount =3D 0;
        return;
    }
=09
```

```
                rootNode.depth=3D0;
                vectorize(rootNode,true,v);
                viewCount =3D v.size();
            }
=09
            private void vectorize
                        (TreeNode    node,
                        boolean      respectExpanded,
                        Vector              nodeVector)
            {
              if (node =3D=3D null)
                return;
            =09
              nodeVector.addElement(node);
            =09
              if ((!respectExpanded && node.child !=3D null) || =
node.isExpanded())
              {
                node.child.depth =3D node.depth + 1;
                vectorize(node.child,respectExpanded,nodeVector);
              }
            =09
              if (node.sibling !=3D null)
              {
                node.sibling.depth =3D node.depth;
                vectorize(node.sibling,respectExpanded,nodeVector);
              }
            }
=09
=09
            public boolean handleEvent(Event event)
```

```
        {
          if (event.target =3D=3D sbV)
          {
                    if (descrip !=3D null)
                    {
                              descrip =3D null;
                              redraw();
                    }
            if (event.arg =3D=3D null)
            {
              return false;
            }
=09
            if (sbVPosition !=3D ((Integer) event.arg).intValue())
            {
              sbVPosition =3D ((Integer) event.arg).intValue();
              redraw();
            }
          }
              //if ((event.target =3D=3D this) && (event.id =3D=3D =
Event.MOUSE_DOWN))
              //        return mouseDown(event, event.x, event.y);
=09
          return(super.handleEvent(event));
              //return false;
        }
=09

    public void dragNode(int currX, int currY)
    {
      Graphics g =3D getGraphics();
```

```
        g.setXORMode(getBackground());
        if (prevX > 0)
                {
          g.drawRect( prevX, prevY, =
    g.getFontMetrics().stringWidth(selectedNode.reason.name) + 4, =
    g.getFontMetrics().getHeight() + 4);
                        //g.drawString(selectedNode.reason.name, prevX+2, =
    g.getFontMetrics().getHeight() + 2);
                }
        prevX =3D currX;
                prevY =3D currY;
                if (currX > 0)
                {
                        g.drawRect( currX, currY, =
    g.getFontMetrics().stringWidth(selectedNode.reason.name) + 4, =
    g.getFontMetrics().getHeight() + 4);
                        //g.drawString(selectedNode.reason.name, currX+2, =
    g.getFontMetrics().getHeight() + 2);
                }
        g.dispose();
    }
    =09
        public boolean mouseDrag(Event event, int x, int y)
        {
                if (selectedNode =3D=3D null)
                        return true;
                if (selectedNode.reason.id < 0)
                {
                        selectedNode.expand();
                        redraw();
                        return true;
```

```
                }
                dragNode(x, y);
                return true;
        }

        public boolean mouseUp(Event event, int x, int y)
        {
                dragNode(-1, -1);
                repaint();
                if (x > size().width)
                        postEvent(new Event(this, 10000, new Integer(100 * y / =
size().height)));
                return true;
        }

        public boolean mouseDown(Event event, int x, int y)
        {
          int index =3D (y/cellSize) + sbVPosition;
=09
          if (index > viewCount-1)
          {
                        descrip =3D null;
                        redraw();
            return false; //clicked below the last node
          }
=09
          TreeNode oldNode =3D selectedNode;
          TreeNode newNode =3D (TreeNode)v.elementAt(index);
          int newDepth =3D newNode.getDepth();
=09
          changeSelection(newNode);
```

```
Rectangle toggleBox =3D new Rectangle(cellSize*newDepth + =
cellSize/4,
                                    (index-sbVPosition)*cellSize + =
clickSize/2,
                                    clickSize, clickSize);
                if (event.modifiers !=3D Event.META_MASK)
                {
                        if (toggleBox.inside(x,y))
                        {
                                newNode.toggle();
                                sendActionEvent(event);
                                redraw();
                        }
                        else
                        {
                                // check for double click
                                long currentTime =3D event.when;
        =09
                                if ((newNode=3D=3DoldNode) && =
((event.when-timeMouseDown)<doubleClickResolution))
                                {
                                        newNode.toggle();
                                        redraw();
                                        sendActionEvent(event);
                                        postEvent(new Event(this, 10000, null));
                                        return false;
                                }
                                else
                                {
                                        //single click action could be added here
                                        timeMouseDown =3D event.when;
```

```
                                }
        =09
                        }
                }
5       =09
                if ((newNode !=3D null) && (event.modifiers =3D=3D
Event.META_MASK))
                {
                        descrip =3D newNode.reason.descrip;
10                      redraw();
                } else {
                        descrip =3D null;
                        redraw();
                }
15      =09
                return true;
        }
=09
        public boolean mouseMove(Event event, int x, int y)
20      {
                return true;
        }
        public boolean keyDown(Event event, int key)
        {
25        int index =3D v.indexOf(selectedNode);
=09
          switch (key)
          {
            case 10:   //enter key
30            sendActionEvent(event);
              requestFocus();
```

```
                    break:
                case Event.LEFT:    //left arrow
                    if (selectedNode.isExpanded())
                    {
                        selectedNode.toggle();
                        redraw();
                        break;
                    }
                    // else drop through to "UP" with no "break;"
                case Event.UP:
                    if (index > 0)
                    {
                        index--;
                        changeSelection((TreeNode)v.elementAt(index));
                        requestFocus();
                    }
                    break;
                case Event.RIGHT:
                    if (selectedNode.isExpandable() && = (!selectedNode.isExpanded()))
                    {
                        selectedNode.toggle();
                        sendActionEvent(event);
                        redraw();
                        break;
                    }
=09
                    if (!selectedNode.isExpandable())
                    {
                        break;
                    }
                    // else drop thru' to DOWN
```

```
                case Event.DOWN:
                    if (index < viewCount-1)
                    {
                        index++;
                        changeSelection((TreeNode)v.elementAt(index));
                        requestFocus();
                    }
                    break;
                }
=09
                return false;
            }
        =09
            private void sendActionEvent(Event event)
            {
                int id =3D event.id;
                Object arg =3D event.arg;
                event.id =3D Event.ACTION_EVENT;
                event.arg =3D new String(selectedNode.getText());
                postEvent(event);
                event.id =3D id;
                event.arg =3D arg;
            }
        =09
            /**
             * Returns the currently selected node.
             */
            public TreeNode getSelectedNode()
            {
                return selectedNode;
            }
```

```
=09
        /**
         * Gets the text of the currently selected node.
         * @return the text of the currently selected node or null if no node
         * is selected
         */
        public String getSelectedText()
        {
          if (selectedNode=3D=3Dnull)
          {
            return null;
          }
=09
          return selectedNode.getText();
        }
=09
        private void changeSelection(TreeNode node)
        {
          if (node =3D=3D selectedNode)
          {
            return;
          }
          TreeNode oldNode =3D selectedNode;
          selectedNode =3D node;
          drawNodeText(oldNode, (v.indexOf(oldNode) - sbVPosition)*cellSize, =
true);
          drawNodeText(node,  (v.indexOf(node)  - sbVPosition)*cellSize, =
true);
          // send select event
=09
          int index =3D v.indexOf(selectedNode);
```

```
=09
            postEvent(new Event(this, SEL_CHANGED, selectedNode));
=09
            if (index < sbVPosition)
            { //scroll up
              sbVPosition--;
              sbV.setValue(sbVPosition);
              redraw();
              return;
            }
=09
            if (index >=3D sbVPosition + (viewHeight-cellSize/2)/cellSize)
            {
              sbVPosition++;
              sbV.setValue(sbVPosition);
              redraw();
              return;
            }
=09
            repaint();
          }
=09
          public void update (Graphics g)
          {
            //(eliminates background draw to reduce flicker)
            paint(g);
          }
=09
          public void paint (Graphics g)
          {
                Dimension d =3D size();
```

```
=09
        if ((d.width !=3D viewWidth) || (d.height !=3D viewHeight))
            redraw();
        g.drawImage(im1, 0, 0, this);
    }
=09
    public void redraw()
    {
      resetVector();
=09
      if (viewCount > viewHeight/cellSize)
      {
        // need the vertical scrollbar
        sbV.setValues(sbVPosition, (viewHeight/cellSize), 0, =
viewCount-2);
            sbV.setPageIncrement(1);
            sbVWidth =3D sbV.preferredSize().width;
            getParent().paintAll(getParent().getGraphics());
        sbV.show();
        layout();
      }
      else
      {
        sbV.hide();
        sbVWidth =3D 1;
        sbVPosition =3D 0;
        layout();
      }
=09
      drawTree();
      repaint();
```

```
        }
=09
  /**
   * Draws the ReasonTree into an offscreen image. This is used for =
cleaner refresh.
   */
        public void drawTree()
        {
                Dimension d =3D size();
=09
                if ((d.width !=3D viewWidth) || (d.height !=3D viewHeight) || =
(g1=3D=3Dnull))
                {
              // size has changed, must resize image
                        im1 =3D createImage(d.width, d.height);
                        if (g1 !=3D null) {
                                g1.dispose();
                        }
                g1 =3D im1.getGraphics();
                viewWidth=3Dd.width;
                viewHeight=3Dd.height;
                }
=09
              Font f =3D getFont();  //unix version might not provide a default =
font
=09
                    //Make certain there is a font
              if (f =3D=3D null)
              {
                f =3D new Font("TimesRoman", Font.PLAIN, 13);
                g1.setFont(f);
```

```
            setFont(f);
        }
=09
        //Make certain the graphics object has a font (Mac doesn't seem to)
            if (f !=3D null)
            {
                if (g1.getFont() =3D=3D null)
                    g1.setFont(f);
            }
=09
        fm =3D g1.getFontMetrics();
        g1.setColor(getBackground());
        g1.fillRect(0,0,viewWidth,viewHeight);  // clear image
=09
        //do drawing for each visible node
        int lastOne=3DsbVPosition+viewHeight/cellSize+1;
=09
        if (lastOne > viewCount)
        {
            lastOne =3D viewCount;
        }
=09
        TreeNode outerNode =3D (TreeNode)v.elementAt(sbVPosition);
        for (int i=3DsbVPosition; i<lastOne; i++)
        {
            TreeNode node=3D(TreeNode)v.elementAt(i);
            int x =3D cellSize*(node.depth + 1);
            int y =3D (i-sbVPosition)*cellSize;
=09
            // draw lines
            g1.setColor(getForeground());
```

```
=09
                // draw vertical sibling line
                if (node.sibling !=3D null)
                {
5                  int k =3D v.indexOf(node.sibling) - i;
=09
                   if (k > lastOne)
                   {
                      k =3D lastOne;
10                 }
=09
                   drawDotLine(x - cellSize/2, y + cellSize/2,
                            x - cellSize/2, y + cellSize/2 +  k*cellSize);
=09
15              }
                // if sibling is above page, draw up to top of page for this =
level
                for (int m=3D0; m<i; m++)
                {
20                 TreeNode sib =3D (TreeNode)v.elementAt(m);
=09
                   if ((sib.sibling =3D=3D node) && (m<sbVPosition))
                   {
                      drawDotLine (x - cellSize/2, 0,
25                             x - cellSize/2, y + cellSize/2);
                   }
                }
=09
                // draw vertical child lines
30              if (node.isExpanded())
                {
```

```
                drawDotLine(x + cellSize/2, y + cellSize -2 ,
                        x + cellSize/2, y + cellSize + cellSize/2);
            }
            // draw node horizontal line
            g1.setColor(getForeground());
            drawDotLine(x - cellSize/2, y + cellSize/2,
                    x + cellSize/2, y + cellSize/2);
=09
            // draw toggle box
            if (node.isExpandable())
            {
              g1.setColor(getBackground());
              g1.fillRect(cellSize*(node.depth) + cellSize/4, y + =
clickSize/2, clickSize, clickSize );
              g1.setColor(getForeground());
              g1.drawRect(cellSize*(node.depth) + cellSize/4, y + =
clickSize/2, clickSize, clickSize );
              // cross hair
              g1.drawLine(cellSize*(node.depth) + cellSize/4 +2,        =
  y + cellSize/2,
                      cellSize*(node.depth) + cellSize/4 + clickSize =
-2, y + cellSize/2);
=09
              if (!(node.isExpanded()))
              {
                g1.drawLine(cellSize*(node.depth) + cellSize/2, y + =
clickSize/2 +2,
                        cellSize*(node.depth) + cellSize/2, y + =
clickSize/2 + clickSize -2);
              }
            }
```

```
=09
            // draw node image
            Image nodeImage =3D node.getImage();
=09
            if (nodeImage !=3D null)
            {
              g1.drawImage(nodeImage, x+imageInset, y, this);
            }
=09
            // draw node text
            if (node.getText() !=3D null)
            {
              drawNodeText(node, y, node=3D=3DselectedNode);
            }
=09
            if(outerNode.depth>node.depth)
              outerNode =3D node;
          }
=09
          // draw outer vertical lines
          while((outerNode =3D outerNode.parent) !=3D null)
          {
            if(outerNode.sibling !=3D null)
              drawDotLine (cellSize*(outerNode.depth + 1) - cellSize/2, =
0,
                  cellSize*(outerNode.depth + 1) - cellSize/2, = d.height);;
          }
        =09

            if (descrip !=3D null)
            {
```

```
                    int width =3D size().width - 20 - sbVWidth;
                =09
                    Vector lines =3D Globals.breakIntoLines(descrip, width, =
g1.getFontMetrics());
                    int y =3D size().height - (lines.size() * =
g1.getFontMetrics().getHeight());
                    g1.setColor(new Color(255, 255, 204));
                    g1.fillRect(0, y - g1.getFontMetrics().getHeight(), size().width,
=
size().height);
                    g1.setColor(getForeground());
                    y +=3D g1.getFontMetrics().getHeight() >> 1;
                    for (int i =3D 0; i < lines.size(); i++)
                    {
                        g1.drawString((String)lines.elementAt(i), 10, y);
                        y +=3D g1.getFontMetrics().getHeight();
                    }
                }
                // draw border
            g1.setColor(getBackground());
                g1.setColor(getForeground());
            g1.drawRect(0,0,viewWidth - sbVWidth, viewHeight - 1);
        }
=09
        private void drawNodeText(TreeNode node, int yPosition, boolean =
eraseBackground)
        {
            Color fg, bg;
            int depth=3Dnode.depth;
            Image nodeImage =3D node.getImage();
            int textOffset =3D ((depth + 1) * (cellSize)) + cellSize + =
```

```
textInset - (nodeImage=3D=3Dnull ? 12:0);
=09
            if (node=3D=3DselectedNode)
            {
               fg=3DfgHighlightColor;
               bg=3DbgHighlightColor;
            }
            else
            {
               fg =3D getForeground();
               bg =3D getBackground();
            }
=09
            if (eraseBackground)
            {
               g1.setColor(bg);
               g1.fillRect(textOffset-1, yPosition+1, =
fm.stringWidth(node.getText())+4, cellSize-1);
            }
=09
            g1.setColor(fg);
            g1.drawString(node.getText(), textOffset, yPosition + cellSize - =
textBaseLine);
         }
=09
         private void drawDotLine(int x0, int y0, int x1, int y1)
         {
           if (y0=3D=3Dy1)
           {
              for (int i =3D x0; i<x1; i+=3D2)
              {
```

```
                    g1.drawLine(i,y0, i, y1);
                  }
                }
                else
                {
                  for (int i =3D y0; i<y1; i+=3D2)
                  {
                    g1.drawLine(x0, i, x1, i);
                  }
                }
              }
=09
=09
=09
              public synchronized Dimension preferredSize()
              {
                return new Dimension(175, 125);
              }
=09
              public synchronized Dimension minimumSize()
              {
                return new Dimension(50, 50);
              }
              public void setLayout(LayoutManager lm)
              {
              }
            }
            class InvalidTreeNodeException
              extends Exception
            {
            }
```

What is claimed is:

1. An electronic interface for permitting a user to communicate subjective data information, the interface comprising:

a parameter menu, said menu providing a user visible set of data parameters which may be associated with the subjective data information;

a parameter canvas, said canvas being simultaneously visible with said parameter menu, and being usable by the user for identifying any of such data parameters that are associated with the subjective data information.

2. The interface of claim 1, wherein all of the user's subjective data information is captured by such interface using said data parameters.

3. The interface of claim 1, wherein all of the user's subjective data information is captured by such interface during the entirety of a data collection session using a single data collection screen.

4. The interface of claim 1, wherein the subjective data information pertains to the user's mental impressions of an actual or proposed action and/or transaction.

5. The interface of claim 5, wherein said transaction is an event or an item of interest to the user.

6. The interface of claim 1, wherein the subjective data information pertains to lessons learned by such user associated with an action and/or transaction.

7. The interface of claim 1, wherein said data parameters associated with the subjective data information are selected and moved by such user along a distance spanning from said parameter menu to said parameter canvas by physically manipulating an electronic pointing device.

8. The interface of claim 7, wherein said distance is less than approximately half the width of said interface as seen by the user.

9. The interface of claim 1, wherein said identified data parameters associated with the subjective data information are stored as one or more electronic records corresponding to an electronic data picture.

10. The interface of claim 9, wherein said data picture includes numeric data values, but is generated without numeric data input by the user.

11. The interface of claim 10, wherein said numeric data values are based on the physical location of said data parameters as placed by the user on said parameter canvas.

12. The interface of claim 11, wherein said electronic data picture can be retrieved and modified at a later time by the user using such interface.

13. The interface of claim 1, wherein said data parameters can be ranked in relative importance by the user based on their location on said parameter canvas.

14. The interface of claim 13, wherein a relative ranking between data parameters can be changed by the user by altering a relative physical arrangement of said data parameters on said parameter canvas.

15. The interface of claim 13, wherein said data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

16. The interface of claim 13, wherein said parameter canvas includes a gradient visible to the user for assisting in the ranking of said data parameters.

17. The interface of claim 16, further wherein said parameter canvas conveys visible feedback information when the user is arranging said data parameters.

18. The interface of claim 1, wherein when the interface is invoked by the user, an initial proposed data picture is presented to the user on said data canvas, which initial proposed data picture can be modified by the user.

19. The interface of claim 1, wherein said data parameters include data parameters visibly displayed in text format.

20. The interface of claim 1, wherein said data parameters include data parameters visibly displayed in symbolic format.

21. The interface of claim 1, wherein said data parameters include factors associated with a user's reasons for performing or engaging in a particular activity.

22. The interface of claim 1, wherein said data parameters include factors associated with a user's mental impressions of an item or event.

23. The interface of claim 1, wherein said data parameters include factors associated with lessons learned by a user concerning an event.

24. An electronic interface for permitting an operator to identify parameters concerning an action and/or transaction, the interface comprising:

a menu providing a set of parameters available for selection by the operator, said menu occupying a first portion of the interface;

a preference field occupying a second portion of the interface for visually displaying any identified personal parameters selected by the operator for the particular operator transaction from said set of parameters;

wherein the operator can dynamically select such personal parameters in said menu and move them to said preference field; and

further wherein said personal parameters are stored in a transaction record form usable by a computing system.

25. The interface of claim 24, wherein the personal parameters describe one or more mental impressions of such transaction.

26. The interface of claim 24, wherein said parameters are selected and moved by such operator along a distance spanning from said menu to said preference field by physically manipulating an electronic pointing device.

27. The interface of claim 26, wherein said distance is less than approximately half the width of said interface as seen by the operator.

28. The interface of claim 24, wherein said transaction record form includes numeric data values, but is generated without numeric data input by the operator.

29. The interface of claim 28, wherein said numeric data values are based on the physical location of said parameters as placed by the operator in said preference field.

30. The interface of claim 24, wherein said identified personal parameters can be ranked in relative importance by the operator based on their location in said preference field.

31. The interface of claim 30, wherein a relative ranking between data parameters can be changed by the user by altering a relative physical arrangement of said data parameters on said parameter canvas.

32. The interface of claim 30, wherein said preference field includes a gradient visible to the operator for assisting in the ranking of said parameters.

33. The interface of claim 30, wherein said parameters can be ranked by both a relative horizontal and vertical location in said preference field.

34. The interface of claim 32, further wherein said preference field conveys visible feedback information when the operator is arranging said parameters.

35. The interface of claim 24, wherein when the interface is invoked by the operator, an initial proposed arrangement of parameters is presented to the operator in said preference field, which initial proposed arrangement can be modified by the operator.

36. The interface of claim 24, wherein said parameters include factors associated with an operator's reasons for performing or engaging in a particular activity.

37. The interface of claim 22, wherein said parameters include factors associated with lessons learned by a user concerning an event.

38. The interface of claim 22, wherein said transaction record can be retrieved and modified by the operator at a later time.

39. The interface of claim 22, wherein said parameters can be customized by the operator.

40. The interface of claim 22, wherein said interface also provides a visual comparison between data in said transaction record and other transaction records.

41. The interface of claim 22, wherein said interface also provides visual feedback to such operator based on an evaluation of said data in said transaction record.

42. An electronic interface for collecting information for a data picture, the interface comprising:

a data palette providing a set of data parameters available for selection; and

a data canvas, separate from said data palette, on which said data parameters can be displayed and arranged arbitrarily by a user to generate the data picture; and

wherein said data picture embodies information collected from the user and pertaining to the user's perceptions concerning a particular action and/or transaction.

43. The interface of claim 42, wherein said data parameters are selected and moved by such user to a gradient on said data canvas by physically manipulating an electronic pointing device.

44. The interface of claim 42, wherein said data picture is generated using a single data capture screen including said data palette and said data canvas.

45. The interface of claim 42, wherein said data picture is translatable into one or more electronic records including numeric data values, but said data picture is generated without numeric data input by the user.

46. The interface of claim 45, wherein said numeric data values are based on the physical location of said data parameters as placed by the user on said data canvas.

47. The interface of claim 42, wherein said data parameters can be ranked in relative importance by the user based on their location on said data canvas.

48. The interface of claim 47, further wherein said data canvas conveys visible feedback information when the user is arranging said data parameters.

49. The interface of claim 42, wherein said data parameters include factors associated with lessons learned by a user concerning such action and/or transaction.

50. The interface of claim 42, wherein said interface also provides a visual comparison between data in said data picture and other data pictures.

51. The interface of claim 42, wherein said interface also provides visual feedback to such operator based on an evaluation of said data in said transaction record.

52. The interface of claim 42, wherein said parameters can be customized by the user.

53. A method of inputting data to a computer program, said method comprising:

presenting a parameter menu to a user. said menu providing a user visible set of data parameters which may be associated with subjective data information;

presenting a parameter canvas to said user, said canvas being simultaneously visible with said parameter menu, and being usable by the user for identifying any of such data parameters that are associated with the subjective data information:

wherein the data input to such computer program consists of said data parameters identified by said user on said parameter canvas.

54. The method of claim 53, wherein the subjective data information pertains to the user's mental impressions of an actual or proposed transaction.

55. The method of claim 54, wherein said transaction is an event or an item of interest to the user.

56. The method of claim 53, wherein said data parameters associated with the subjective data information can be selected and moved by said user along a distance spanning from said parameter menu to said parameter canvas by physically manipulating an electronic pointing device.

57. The method of claim 56, wherein said distance is less than approximately half the width of said interface as seen by the user.

58. The method of claim 53, further including a step of storing said data parameters as one or more electronic records corresponding to an electronic data picture.

59. The method of claim 58, wherein said data picture includes numeric data values, but is generated without numeric data input by the user.

60. The method of claim 59, wherein said numeric data values are based on the physical location of said data parameters as placed by the user on said parameter canvas.

61. The method of claim 53, further including a step of ranking said data parameters on said parameter canvas.

62. The method of claim 61, wherein said data parameters can be ranked according to their physical arrangement on said parameter canvas.

63. The method of claim 61, further including a step of providing a gradient visible to the user for assisting in the ranking of said data parameters.

64. The method of claim 61, further including a step of providing visible feedback information when the user arranges said data parameters.

65. The method of claim 61, wherein said data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

66. The method of claim 53, further including a step of presenting an initial data picture to the user on said data canvas, which initial proposed data picture can be modified by the user.

67. The method of claim 53, wherein said data parameters include factors associated with a user's reasons for performing or engaging in a particular activity.

68. The method of claim 53, wherein said data parameters include factors associated with lessons learned by a user concerning an event.

69. The method of claim 53, further including a step of providing a visual comparison between said data input and other data previously input using said parameter canvas.

70. The method of claim 53, further including a step of providing visual feedback based on an evaluation of said data input.

71. The method of claim 53, wherein said parameter canvas captures substantially all of the user's subjective data information.

72. The method of claim 53, wherein all of the user's subjective data information is captured during a data collection session using a single data collection screen.

73. The method of claim 53, wherein said parameters can be customized by the operator.

74. The method of claim 53, wherein said data input by said user is utilized by part of an applications program executable by said user using a computing system.

75. A method for permitting a user to identify personal parameters concerning an action and/or transaction to a computer program, the method comprising the steps of:

providing a menu of one or more parameters available for selection by the user, said menu being displayed in a first portion of an interface visible to the user; and

providing a preference field occupying a second portion of the interface visible to the user; and

permitting the user to move any of said parameters to said preference field so as to identify such user's personal parameters associated with the transaction; and

generating a data picture by visually displaying said personal parameters ranked in an order of importance to the user; and

storing said personal parameters are stored in a transaction record form usable by the computer program.

76. The method of claim 75, wherein all of the user's personal parameters are captured using a single data picture.

77. The method of claim 75, wherein all of the user's personal parameters are captured during a data collection session using a single data collection screen.

78. The method of claim 75, wherein the personal parameters describe one or more mental impressions of such action and/or transaction.

79. The method of claim 75, wherein said parameters associated with the transaction can be selected and moved by said user along a distance spanning from said menu to said preference field by physically manipulating an electronic pointing device.

80. The method of claim 79, wherein said distance is less than approximately half the width of said interface as seen by the user.

81. The method of claim 75, wherein said transaction record includes numeric data values, but is generated without numeric data input by the user.

82. The method of claim 81, wherein said numeric data values are based on the physical location of said parameters as placed by the user in said preference field.

83. The method of claim 75, wherein said parameters can be ranked according to their physical arrangement in said preference field.

84. The method of claim 75, further including a step of providing a gradient visible to the user for assisting in the ranking of said parameters.

85. The method of claim 75, further including a step of providing visible feedback information when the user arranges said parameters.

86. The method of claim 75, wherein said data parameters can be ranked by both a relative horizontal and vertical location in said preference field.

87. The method of claim 75, further including a step of presenting an initial data picture to the user in said preference fields, said initial data picture being based on prior data pictures previously entered by the user.

88. The method of claim 75, wherein said parameters include factors associated with a user's reasons for performing or engaging in a particular activity.

89. The method of claim 75, further including a step of providing a visual comparison between said data picture and other data pictures.

90. The method of claim 75, further including a step of providing visual feedback based on an evaluation of said data input.

91. A method of generating a data picture using a computer program, the method comprising the steps of:

providing a data palette, said palette including a set of data parameters available for selection by a user of the program; and

providing a data canvas, separate from said data palette, on which said data parameters can be displayed and arranged arbitrarily by said user to generate the data picture; and

wherein said data picture embodies information collected from said user and pertaining to the user's mental impressions concerning a particular action and/or transaction.

92. The method of claim 91, wherein all of the information collected from said user is captured using a single data picture.

93. The method of claim 91, wherein all of the user's information is captured during a data collection session using a single data collection screen.

94. The method of claim 91, wherein said data picture is stored as part of a transaction record which includes numeric data values, but said data picture is generated without numeric data input by the user.

95. The method of claim 91, wherein said numeric data values are based on the physical location of said data parameters as placed by the user on said data canvas.

96. The method of claim 91, further including a step of permitting said user to rank said data parameters on said data canvas.

97. The method of claim 91, wherein said data parameters can be ranked according to their physical arrangement on said data canvas.

98. The method of claim 91, further including a step of providing visual feedback based on an evaluation of said data input.

99. A method of permitting a user to input a data picture expressing mental impressions concerning an action and/or transaction, the method comprising the steps of:

providing a set of a plurality of individual assertions, said assertions being associated with such mental impressions: and

displaying said set of assertions to the user in a first portion of a visible electronic interface; and

permitting the user to select and move personalized individual assertions taken from said set of assertions to a second, separate portion of said visible interface, which separate portion acts as a data canvas for displaying such personalized individual assertions; and

wherein said personalized individual assertions can be arranged by the user to create the data picture.

100. The method of claim 99, wherein all of the information collected from said user is captured using a single data picture.

101. The method of claim 99, wherein all of the user's information is captured during a data collection session using a single data collection screen.

102. The method of claim 99, wherein numeric data values are assigned to said data parameters based on the physical location of said data parameters as placed by the user on said data canvas.

103. The method of claim 99, further including a step of permitting said user to rank said data parameters on said data canvas.

104. The method of claim 103, wherein said data parameters can be ranked according to their physical arrangement on said data canvas.

105. The method of claim 99, further including a step of providing visual feedback based on an evaluation of said data input.

106. A method of capturing data concerning an actual or proposed transaction from a user of a computing system, said system including at least a keyboard and pointing device for inputting data, the method comprising the steps of:

providing a set of a plurality of individual assertions, said assertions being associated with mental impressions of the user relating to the transaction; and

displaying said set of assertions to the user in a first portion of a visible electronic interface; and

permitting the user to select and move individual ones of said assertions taken from said set of assertions to a second, separate portion of said visible interface, which separate portion acts to visibly display such selected individual assertions along a gradient; and

permitting the user to arrange said selected individual assertions in a ranking order relative to each other along said gradient;

wherein said data is collected from said user substantially without input from the keyboard, and said data is calculated based only on those selected individual assertions from the user.

107. The method of claim 106 further wherein all of the information collected from said user is captured using said set of assertions.

108. The method of claim 106 further wherein all of the user's information is captured during a data collection session using a single data collection screen.

109. The method of claim 106, wherein numeric data values are assigned to said selected individual assertions based on their physical location as placed by the user on said data canvas.

110. The method of claim 106, further including a step of providing a visual comparison between said data and data collected from said user during a prior data capture session.

111. A method of generating program data from user input data concerning an actual or proposed action and/or transaction, the method comprising the steps of:

providing the user with a palette of individual data parameters associated with the user's perceptions of such action and/or transaction; and

permitting the user to select and move individual ones of said assertions taken from said set of assertions to a second, separate portion of said visible interface, which separate portion acts to visibly display such selected individual assertions; and

permitting the user to arrange said selected individual assertions in a ranking order relative to each other so as to constitute user input data;

converting said user input data into program data, by assigning numerical values to such program data corresponding to said arrangement of said selected individual assertions.

112. The method of claim 111, wherein said numeric data values are based on the physical location of said individual assertions as placed by the user on said second portion of said interface.

113. The method of claim 111, further including a step of providing a gradient visible to the user for assisting in the ranking of said individual assertions.

114. The method of claim 111, further including a step of providing visible feedback information when the user arranges said individual assertions.

115. The method of claim 111, wherein said individual assertions include statements associated with lessons learned by a user concerning such action and/or transaction.

116. The method of claim 116, further including a step of retrieving and modifying said lessons at a later time.

117. The method of claim 111 wherein said assertions can be customized by the user.

118. The method of claim 111, further including a step of providing a visual comparison between said data and program data collected from said user during a prior session.

119. The method of claim 111, further including a step of providing visual feedback based on an evaluation of said program data.

120. The method of claim 111 further wherein all of the user's information is captured during a data collection session using a single data collection screen.

Fig. 1

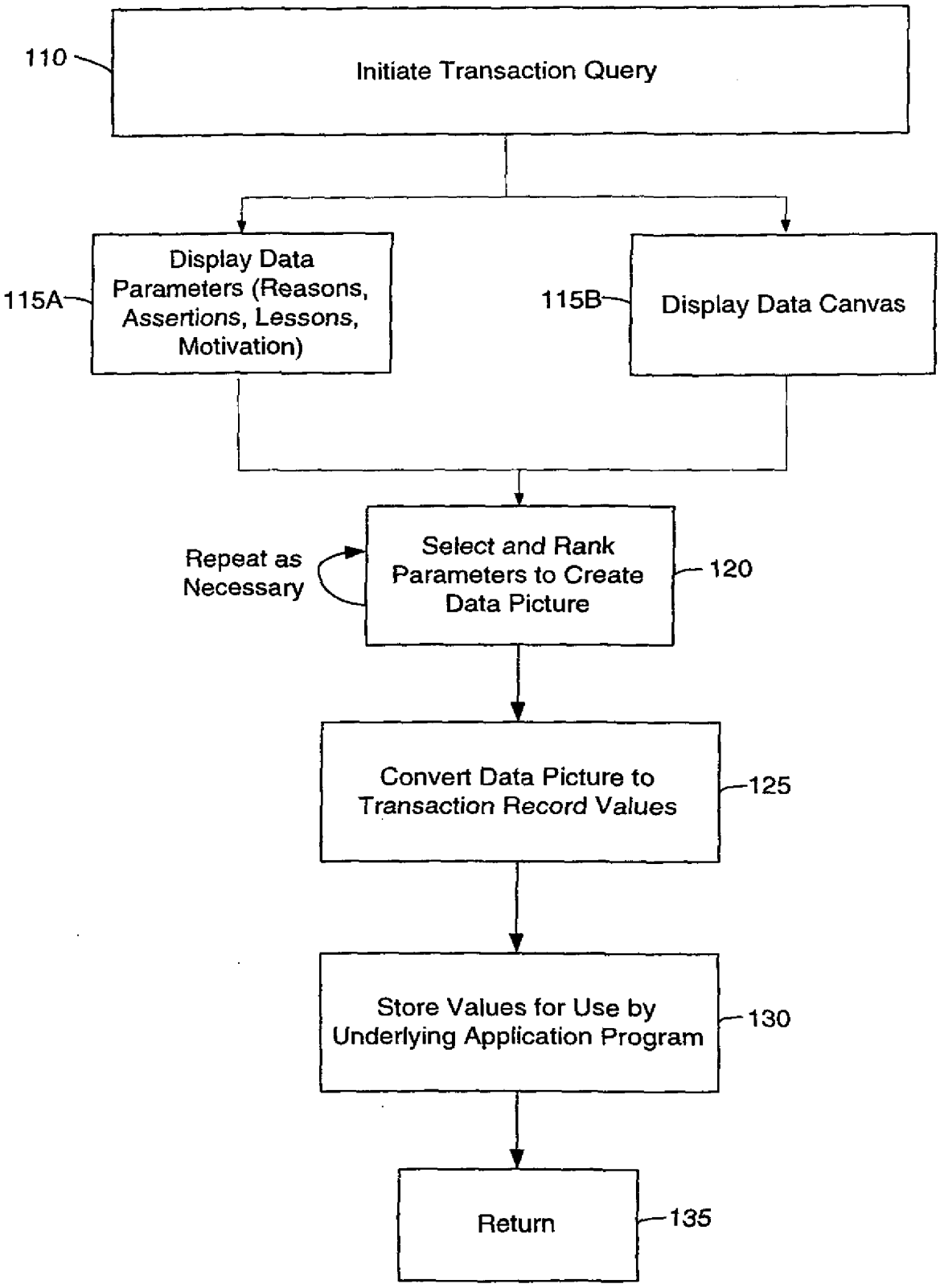
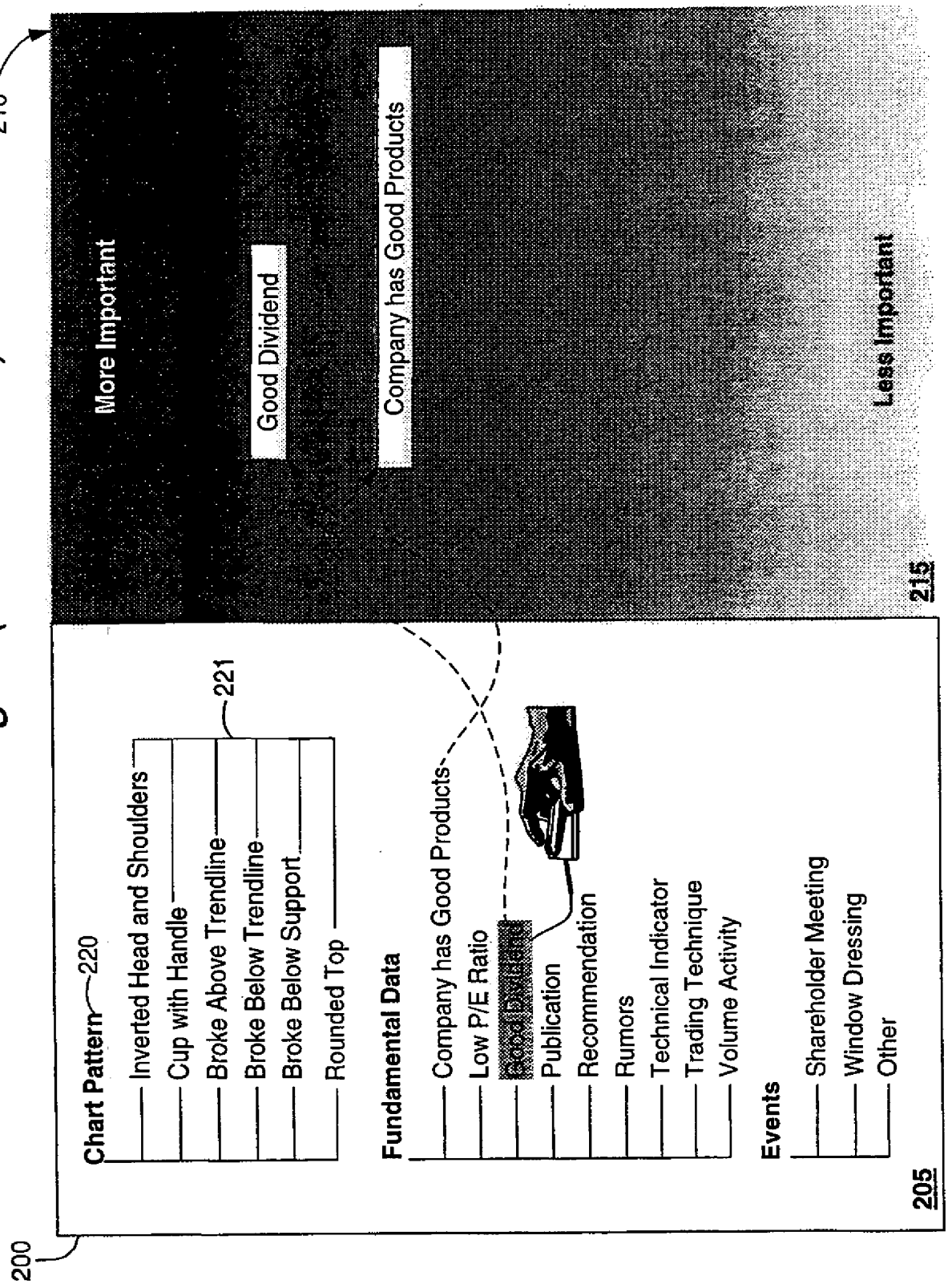

110
Initiate Transaction Query
115A
Display Data Parameters (Reasons, Assertions, Lessons, Motivation)
115B
Display Data Canvas
Repeat as Necessary
Select and Rank Parameters to Create Data Picture
120
Convert Data Picture to Transaction Record Values
125
Store Values for Use by Underlying Application Program
130
Return
135

Fig. 2 (Sheet 1 of 2)



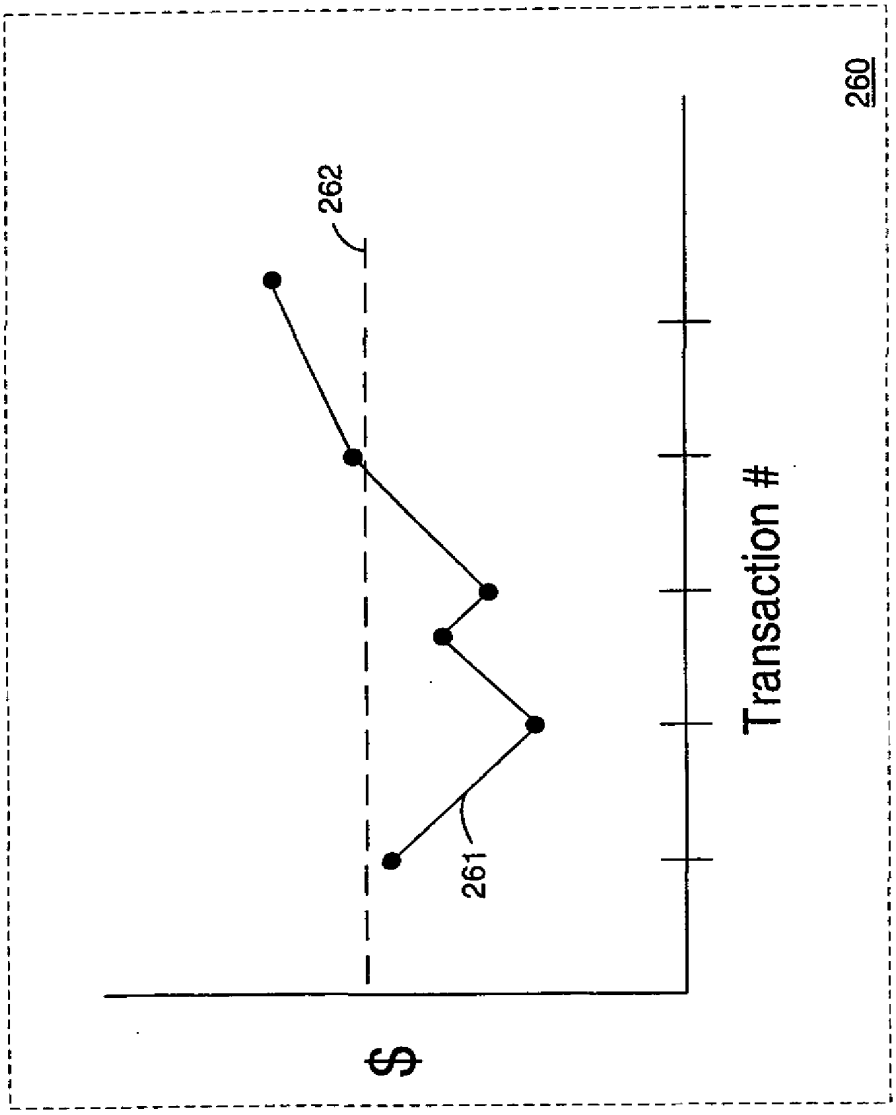


Fig. 2 (Sheet 2 of 2)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24369

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 3/14
US CL :345/352

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 345/146, 333-335, 350-354, 356
707/101-104, 906, 908

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,668,996 A (ONO et al) 16 September 1997, especially col. 3, lines 1 - 25.	1-24 and 53-74
A, P	US 5,923,327 A (SMITH et al) 13 July 1999	1-24 and 53 - 74
A, E	US 5,999,177 A (MARTINEZ et al) 07 December 1999	1-24 and 53 - 74

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 15 FEBRUARY 2000	Date of mailing of the international search report 02 MAR 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer THOMAS JOSEPH Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)★

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION CONCERNING SUBMISSION OR TRANSMITTAL OF PRIORITY DOCUMENT

(PCT Administrative Instructions, Section 411)

To:

GROSS, John, Nicholas
Suite B
10950 N. Blaney
Cupertino, CA 95014
ETATS-UNIS D'AMERIQUE

Date of mailing (day/month/year) 13 August 2001 (13.08.01)	
Applicant's or agent's file reference PROP98002PCT	**IMPORTANT NOTIFICATION**
International application No. PCT/US99/24369	International filing date (day/month/year) 18 October 1999 (18.10.99)
International publication date (day/month/year) 27 April 2000 (27.04.00)	Priority date (day/month/year) 16 October 1998 (16.10.98)
Applicant PROPHET FINANCIAL SYSTEMS, INC. et al	

1. The applicant is hereby notified of the date of receipt (except where the letters "NR" appear in the right-hand column) by the International Bureau of the priority document(s) relating to the earlier application(s) indicated below. Unless otherwise indicated by an asterisk appearing next to a date of receipt, or by the letters "NR", in the right-hand column, the priority document concerned was submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b).

2. This updates and replaces any previously issued notification concerning submission or transmittal of priority documents.

3. An **asterisk(*)** appearing next to a date of receipt, in the right-hand column, denotes a priority document submitted or transmitted to the International Bureau but not in compliance with Rule 17.1(a) or (b). In such a case, **the attention of the applicant is directed** to Rule 17.1(c) which provides that no designated Office may disregard the priority claim concerned before giving the applicant an *opportunity, upon entry into* the national phase, to furnish the priority document within a time limit which is reasonable under the circumstances.

4. The **letters "NR"** appearing in the right-hand column denote a priority document which was not received by the International Bureau or which the applicant did not request the receiving Office to prepare and transmit to the International Bureau, as provided by Rule 17.1(a) or (b), respectively. In such a case, **the attention of the applicant is directed** to Rule 17.1(c) which provides that no designated Office may disregard the priority claim concerned before giving the applicant an opportunity, upon entry into the national phase, to furnish the priority document within a time limit which is reasonable under the circumstances.

Priority date	Priority application No.	Country or regional Office or PCT receiving Office	Date of receipt of priority document
16 Octo 1998 (16.10.98)	09/173,853	US	18 June 2001 (18.06.01)

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No. (41-22) 740.14.35

Authorized officer

Sean Taylor

Telephone No. (41-22) 338.83.38

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION CONCERNING THE FILING OF AMENDMENTS OF THE CLAIMS

(PCT Administrative Instructions, Section 417)

To:

GROSS, John, Nicholas
Suite B
10950 N. Blaney
Cupertino, CA 95014
ETATS-UNIS D'AMERIQUE

Date of mailing (day/month/year) 11 May 2000 (11.05.00)	
Applicant's or agent's file reference PROP98002PCT	**IMPORTANT NOTIFICATION**
International application No. PCT/US99/24369	**International filing date** (day/month/year) 18 October 1999 (18.10.99)
Applicant PROPHET FINANCIAL SYSTEMS, INC. et al	

1. The applicant is hereby notified that amendments to the claims under Article 19 were received by the International Bureau on:

   09 May 2000 (09.05.00)

2. This date is after the expiration of the time limit under Rule 46.1.

   Consequently, the amendments will not be published and will not be considered for the international procedure.

3. The applicant is reminded that the international application (description, claims and drawings) may be amended during the international preliminary examination under Chapter II, according to Article 34, and in any case, before each of the designated Offices, according to Article 28 and Rule 52, or before each of the elected Offices, according to Article 41 and Rule 78.

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No.: (41-22) 740.14.35

Authorised officer

Yolaine CUSSAC

Telephone No.: (41-22) 338.83.38

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of: **Prophet Financial Systems**

PCT APP. No.: **PCT/US99/24369**

PCT Filing Date: **18 October 1999**

Title: **Graphical Data User Interface**

Authorized Officer: **Joseph**

Atty Docket No.:**PROP98002PCT**

ARTICLE 19 AMENDMENTS

International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Enclosed please find five (5) replacement sheets for the amended claims. The differences between the original sheets and the amended claims are described below.

Claims 1, 2, 3, 5, 7 - 10, 13-18, and 53, 56, 58 - 64, and 71 - 73 are replaced by amended claims bearing the same numbers.

Claims 4, 6, 11, 12, 19 – 23, 54-55, 57, 65 – 70 and 74 are unchanged.

Claims 1, 2, 3, 5, 7 - 10, 13-18, and 53, 56, 58 - 64, and 71 – 73 are amended as follows. The underlined text indicates language added to the claim, and bracketed text indicates language deleted from the claim.

1. An electronic interface for permitting a user to communicate subjective data information, the interface comprising:

a parameter menu, said menu providing a user visible set of data parameters which may be associated with the subjective data information;

a parameter canvas, said canvas being simultaneously visible with said parameter menu, and being usable by the user for identifying <u>and arranging</u> [any of] <u>selected ones of</u> [such] <u>said user visible set of</u> data parameters <u>into a visual form expressing the subjective data information as a graphical arrangement of said selected ones of said visible set of data parameters within said parameter canvas</u> [that are associated with the subjective data information].

2. The interface of claim 1, wherein all of the user's subjective data information concerning an action and/or transaction is captured by such interface in a single graphical arrangement using said data parameters.

3. The interface of claim 1, wherein all of the user's subjective data information concerning an action and/or transaction is captured by such interface during the entirety of a data collection session for said action and/or transaction using a single data collection screen.

5. The interface of claim [5] 4, wherein said transaction is an event or an item of interest to the user.

7. The interface of claim 1, wherein said parameter menu and a location on said parameter canvas for arranging said selected ones are physically separated by a first distance, and said data parameters associated with the subjective data information are selected and moved by such user along [a] said first distance [spanning from said parameter menu to said parameter canvas] by physically manipulating an electronic pointing device.

8. The interface of claim 7, wherein said first distance is less than approximately half the width of said interface as seen by the user.

9. The interface of claim 1, wherein said graphical arrangement of said selected ones of said visible set of [identified] data parameters associated with the subjective data information are stored as one or more electronic records corresponding to an electronic data picture.

10. The interface of claim 9, wherein said electronic data picture includes numeric data values, but is generated entirely in visual form and without numeric data input by the user.

13. The interface of claim 1, wherein said selected ones of said visible set of data parameters can be ranked in relative importance by the user based on their location within said graphical arrangement on said parameter canvas.

14. The interface of claim 13, wherein a relative ranking between said selected ones of said visible set of data parameters can be changed by the user by altering a relative physical arrangement of [said] such data parameters on said parameter canvas.

15. The interface of claim 13, wherein said selected ones of said visible set of data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

16. The interface of claim 13, wherein said parameter canvas includes a gradient visible to the user for assisting in the ranking of said selected ones of said visible set of data parameters.

17. The interface of claim 16, further wherein said parameter canvas conveys visible feedback information when the user is arranging said selected ones of said visible set of data parameters.

18. The interface of claim 1, wherein when the interface is invoked by the user, an initial proposed data picture comprised of a graphical arrangement of one or more of said user visible set of data parameters is presented to the user on said data canvas, which initial proposed data picture can be modified by the user.

53. A method of inputting data to a computer program, said method comprising:

presenting a parameter menu to a user, said menu providing a user visible set of data parameters which may be associated with subjective data information;

presenting a parameter canvas to said user, said canvas being simultaneously visible with said parameter menu, and being usable by the user for expressing the subjective data information in visual form by identifying in a graphical arrangement any of such data parameters that are associated with the subjective data information;

wherein the data input to such computer program consists of data relating to said graphical arrangement of any of such [said] data parameters [identified] arranged by said user on said parameter canvas.

56. The method of claim 53, wherein said data parameters [associated with the subjective data information] to be included in said graphical arrangement can be selected and moved by said user along a distance spanning from said parameter menu to said parameter canvas by physically manipulating an electronic pointing device.

58. The method of claim 53, further including a step of storing said graphical arrangement of said data parameters as one or more electronic records corresponding to an electronic data picture.

59. The method of claim 58, wherein said data picture includes numeric data values, but is generated entirely in visual form and without numeric data input by the user.

60. The method of claim 59, wherein said numeric data values are based on the physical location of any of such [said] data parameters [as] placed by the user on said parameter canvas.

61. The method of claim 53, further including a step of ranking [said] any of such data parameters placed on said parameter canvas by adjusting a spatial location of said data parameters in said graphical arrangement.

62. The method of claim 61, wherein [said] any of such data parameters placed on said parameter canvas can be ranked according to their relative physical arrangement to other data parameters on said parameter canvas.

63. The method of claim 61, further including a step of providing a gradient visible to the user for assisting in the ranking of any of such [said] data parameters placed on said parameter canvas.

64. The method of claim 61, further including a step of providing visible feedback information when the user arranges said data parameters on said parameter canvas.

71. The method of claim 53, wherein said parameter canvas captures substantially all of the user's subjective data information concerning an action and/or transaction.

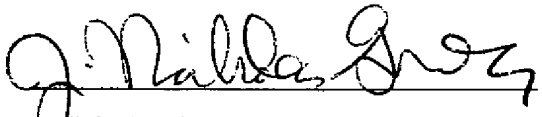
72. The method of claim 53, wherein all of the user's subjective data information concerning an action and/or transaction is captured during a data collection session using a single data collection screen.

73. The method of claim 53, wherein said parameters can be customized by the [operator] user.

Entry of the above is respectfully requested. Should there be any questions about this material, please call the undersigned at 415-551-8298.

Respectfully submitted,

May 2, 2000



J. Nicholas Gross, Esq., Reg. No. 34175
Attorney for Applicant

What is claimed is:

1. An electronic interface for permitting a user to communicate subjective data information, the interface comprising:

a parameter menu, said menu providing a user visible set of data parameters which may be associated with the subjective data information;

a parameter canvas, said canvas being simultaneously visible with said parameter menu, and being usable by the user for identifying and arranging selected ones of said user visible set of data parameters into a visual form expressing the subjective data information as a graphical arrangement of said selected ones of said visible set of data parameters within said parameter canvas.

2. The interface of claim 1, wherein all of the user's subjective data information concerning an action and/or transaction is captured by such interface in a single graphical arrangement using said data parameters.

3. The interface of claim 1, wherein all of the user's subjective data information concerning an action and/or transaction is captured by such interface during the entirety of a data collection session for said action and/or transaction using a single data collection screen.

4. The interface of claim 1, wherein the subjective data information pertains to the user's mental impressions of an actual or proposed action and/or transaction.

5. The interface of claim 4, wherein said transaction is an event or an item of interest to the user.

6. The interface of claim 1, wherein the subjective data information pertains to lessons learned by such user associated with an action and/or transaction.

7. The interface of claim 1, wherein said parameter menu and a location on said parameter canvas for arranging said selected ones are physically separated by a first distance, and said data parameters associated with the subjective data information are selected and moved by such user along said first distance by physically manipulating an electronic pointing device.

8. The interface of claim 7, wherein said first distance is less than approximately half the width of said interface as seen by the user.

9. The interface of claim 1, wherein said graphical arrangement of said selected ones of said visible set of data parameters associated with the subjective data information are stored as one or more electronic records corresponding to an electronic data picture.

10. The interface of claim 9, wherein said electronic data picture includes numeric data values, but is generated entirely in visual form and without numeric data input by the user.

11. The interface of claim 10, wherein said numeric data values are based on the physical location of said data parameters as placed by the user on said parameter canvas.

12. The interface of claim 11, wherein said electronic data picture can be retrieved and modified at a later time by the user using such interface.

13. The interface of claim 1, wherein said selected ones of said visible set of data parameters can be ranked in relative importance by the user based on their location within said graphical arrangement on said parameter canvas.

14. The interface of claim 13, wherein a relative ranking between said selected ones of said visible set of data parameters can be changed by the user by altering a relative physical arrangement of such data parameters on said parameter canvas.

15. The interface of claim 13, wherein said selected ones of said visible set of data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

16. The interface of claim 13, wherein said parameter canvas includes a gradient visible to the user for assisting in the ranking of said selected ones of said visible set of data parameters.

17. The interface of claim 16, further wherein said parameter canvas conveys visible feedback information when the user is arranging said selected ones of said visible set of data parameters.

18. The interface of claim 1, wherein when the interface is invoked by the user, an initial proposed data picture comprised of a graphical arrangement of one or more of said user visible set of data parameters is presented to the user on said data canvas, which initial proposed data picture can be modified by the user.

19. The interface of claim 1, wherein said data parameters include data parameters visibly displayed in text format.

20. The interface of claim 1, wherein said data parameters include data parameters visibly displayed in symbolic format.

21. The interface of claim 1, wherein said data parameters include factors associated with a user's reasons for performing or engaging in a particular activity.

22. The interface of claim 1, wherein said data parameters include factors associated with a user's mental impressions of an item or event.

23. The interface of claim 1, wherein said data parameters include factors associated with lessons learned by a user concerning an event.

53. A method of inputting data to a computer program, said method comprising:

presenting a parameter menu to a user, said menu providing a user visible set of data parameters which may be associated with subjective data information;

presenting a parameter canvas to said user, said canvas being simultaneously visible with said parameter menu, and being usable by the user for expressing the subjective data information in visual form by identifying in a graphical arrangement any of such data parameters that are associated with the subjective data information;

wherein the data input to such computer program consists of data relating to said graphical arrangement of any of such data parameters arranged by said user on said parameter canvas.

54. The method of claim 53, wherein the subjective data information pertains to the user's mental impressions of an actual or proposed transaction.

55. The method of claim 54, wherein said transaction is an event or an item of interest to the user.

56. The method of claim 53, wherein said data parameters to be included in said graphical arrangement can be selected and moved by said user along a distance spanning from said parameter menu to said parameter canvas by physically manipulating an electronic pointing device.

57. The method of claim 56, wherein said distance is less than approximately half the width of said interface as seen by the user.

58. The method of claim 53, further including a step of storing said graphical arrangement of said data parameters as one or more electronic records corresponding to an electronic data picture.

59. The method of claim 58, wherein said data picture includes numeric data values, but is generated entirely in visual form and without numeric data input by the user.

60. The method of claim 59, wherein said numeric data values are based on the physical location of any of such data parameters placed by the user on said parameter canvas.

61. The method of claim 53, further including a step of ranking any of such data parameters placed on said parameter canvas by adjusting a spatial location of said data parameters in said graphical arrangement.

62. The method of claim 61, wherein any of such data parameters placed on said parameter canvas can be ranked according to their relative physical arrangement to other data parameters on said parameter canvas.

63. The method of claim 61, further including a step of providing a gradient visible to the user for assisting in the ranking of any of such data parameters placed on said parameter canvas.

64. The method of claim 61, further including a step of providing visible feedback information when the user arranges said data parameters on said parameter canvas.

65. The method of claim 61, wherein said data parameters can be ranked by both a relative horizontal and vertical location on said parameter canvas.

66. The method of claim 53, further including a step of presenting an initial data picture to the user on said data canvas, which initial proposed data picture can be modified by the user.

67. The method of claim 53, wherein said data parameters include factors associated with a user's reasons for performing or engaging in a particular activity.

68. The method of claim 53, wherein said data parameters include factors associated with lessons learned by a user concerning an event.

69. The method of claim 53, further including a step of providing a visual comparison between said data input and other data previously input using said parameter canvas.

70. The method of claim 53, further including a step of providing visual feedback based on an evaluation of said data input.

71. The method of claim 53, wherein said parameter canvas captures substantially all of the user's subjective data information concerning an action and/or transaction.

72. The method of claim 53, wherein all of the user's subjective data information concerning an action and/or transaction is captured during a data collection session using a single data collection screen.

73. The method of claim 53, wherein said parameters can be customized by the user.

74. The method of claim 53, wherein said data input by said user is utilized by part of an applications program executable by said user using a computing system.

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTICE INFORMING THE APPLICANT OF THE COMMUNICATION OF THE INTERNATIONAL APPLICATION TO THE DESIGNATED OFFICES

(PCT Rule 47.1(c), first sentence)

To:
GROSS, John, Nicholas
Suite B
10950 N. Blaney
Cupertino, CA 95014
ETATS-UNIS D'AMERIQUE

Date of mailing (day/month/year) 27 April 2000 (27.04.00)	
Applicant's or agent's file reference PROP98002PCT	**IMPORTANT NOTICE**

International application No.	**International filing date** (day/month/year)	**Priority date** (day/month/year)
PCT/US99/24369	18 October 1999 (18.10.99)	16 October 1998 (16.10.98)

Applicant
PROPHET FINANCIAL SYSTEMS, INC. et al

1. Notice is hereby given that the International Bureau has communicated, as provided in Article 20, the international application to the following designated Offices on the date indicated above as the date of mailing of this Notice:
   CN,JP,US

   In accordance with Rule 47.1(c), third sentence, those Offices will accept the present Notice as conclusive evidence that the communication of the international application has duly taken place on the date of mailing indicated above and no copy of the international application is required to be furnished by the applicant to the designated Office(s).

2. The following designated Offices have waived the requirement for such a communication at this time:
   CA,DE,EA,EP,GB,MX,RU

   The communication will be made to those Offices only upon their request. Furthermore, those Offices do not require the applicant to furnish a copy of the international application (Rule 49.1(a-bis)).

3. Enclosed with this Notice is a copy of the international application as published by the International Bureau on
   27 April 2000 (27.04.00) under No. WO 00/23872

REMINDER REGARDING CHAPTER II (Article 31(2)(a) and Rule 54.2)

If the applicant wishes to postpone entry into the national phase until 30 months (or later in some Offices) from the priority date, **a demand for international preliminary examination** must be filed with the competent International Preliminary Examining Authority before the expiration of 19 months from the priority date.

It is the applicant's sole responsibility to monitor the 19-month time limit.

Note that only an applicant who is a national or resident of a PCT Contracting State which is bound by Chapter II has the right to file a demand for international preliminary examination.

REMINDER REGARDING ENTRY INTO THE NATIONAL PHASE (Article 22 or 39(1))

If the applicant wishes to proceed with the international application in the **national phase**, he must, within 20 months or 30 months, or later in some Offices, perform the acts referred to therein before each designated or elected Office.

For further important information on the time limits and acts to be performed for entering the national phase, see the Annex to Form PCT/IB/301 (Notification of Receipt of Record Copy) and Volume II of the PCT Applicant's Guide.

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No. (41-22) 740.14.35

Authorized officer
J. Zahra

Telephone No. (41-22) 338.83.38

PATENT COOPERATION TREATY

From the INTERNATIONAL SEARCHING AUTHORITY

To: JOHN NICHOLAS GROSS
10950 N BLANEY
SUITE B
CUPERTINO, CA 95014

PCT

NOTIFICATION OF TRANSMITTAL OF THE INTERNATIONAL SEARCH REPORT OR THE DECLARATION

(PCT Rule 44.1)

Applicant's or agent's file reference PROP98002PCT	Date of Mailing *(day/month/year)* 02 MAR 2000 **FOR FURTHER ACTION** **See paragraphs 1 and 4 below**
International application No. PCT/US99/24369	International filing date *(day/month/year)* 18 OCTOBER 1999
Applicant PROPHET FINANCIAL SYSTEMS, INC.	

1. [X] The applicant is hereby notified that the international search report has been established and is transmitted herewith.

Filing of amendments and statement under Article 19:
The applicant is entitled, if he so wishes, to amend the claims of the international application (see Rule 46):

When? The time limit for filing such amendments is normally 2 months from the date of transmittal of the international search report; however, for more details, see the notes on the accompanying sheet.

Where? Directly to the International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland
Facsimile No.: (41-22) 740.14.35

For more detailed instructions, see the notes on the accompanying sheet.

2. [ ] The applicant is hereby notified that no international search report will be established and that the declaration under Article 17(2)(a) to that effect is transmitted herewith.

3. [ ] **With regard to the protest** against payment of (an) additional fee(s) under Rule 40.2, the applicant is notified that:

[ ] the protest together with the decision thereon has been transmitted to the International Bureau together with the applicant's request to forward the texts of both the protest and the decision thereon to the designated Offices.

[ ] no decision has been made yet on the protest; the applicant will be notified as soon as a decision is made.

4. **Further action(s):** The applicant is reminded of the following:

Shortly after **18 months** from the priority date, the international application will be published by the International Bureau. If the applicant wishes to avoid or postpone publication, a notice of withdrawal of the international application, or of the priority claim, must reach the International Bureau as provided in rules 90 *bis* 1 and 90 *bis* 3, respectively, before the completion of the technical preparations for international publication.

Within **19 months** from the priority date, a demand for international preliminary examination must be filed if the applicant wishes to postpone the entry into the national phase until 30 months from the priority date (in some Offices even later).

Within **20 months** from the priority date, the applicant must perform the prescribed acts for entry into the national phase before all designated Offices which have not been elected in the demand or in a later election within 19 months from the priority date or could not be elected because they are not bound by Chapter II.

Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	Authorized officer THOMAS JOSEPH
Facsimile No. (703) 305-3230	Telephone No. (703) 305-9600

Form PCT/ISA/220 (January 1994)★ *(See notes on accompanying sheet)*

PATENT COOPERATION TREATY

From the INTERNATIONAL SEARCHING AUTHORITY

To: JOHN NICHOLAS GROSS
10950 N BLANEY
SUITE B
CUPERTINO, CA 95014

PCT

NOTIFICATION OF TRANSMITTAL OF THE INTERNATIONAL SEARCH REPORT OR THE DECLARATION

(PCT Rule 44.1)

Date of Mailing *(day/month/year)*	02 MAR 2000
Applicant's or agent's file reference PROP98002PCT	**FOR FURTHER ACTION** **See paragraphs 1 and 4 below**
International application No. PCT/US99/24369	International filing date *(day/month/year)* 18 OCTOBER 1999
Applicant PROPHET FINANCIAL SYSTEMS, INC.	

1. [X] The applicant is hereby notified that the international search report has been established and is transmitted herewith.

Filing of amendments and statement under Article 19:
The applicant is entitled, if he so wishes, to amend the claims of the international application (see Rule 46):

When? The time limit for filing such amendments is normally 2 months from the date of transmittal of the international search report; however, for more details, see the notes on the accompanying sheet.

Where? Directly to the International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland
Facsimile No.: (41-22) 740.14.35

For more detailed instructions, see the notes on the accompanying sheet.

2. [ ] The applicant is hereby notified that no international search report will be established and that the declaration under Article 17(2)(a) to that effect is transmitted herewith.

3. [ ] **With regard to the protest** against payment of (an) additional fee(s) under Rule 40.2, the applicant is notified that:

[ ] the protest together with the decision thereon has been transmitted to the International Bureau together with the applicant's request to forward the texts of both the protest and the decision thereon to the designated Offices.

[ ] no decision has been made yet on the protest; the applicant will be notified as soon as a decision is made.

4. **Further action(s):** The applicant is reminded of the following:

Shortly after **18 months** from the priority date, the international application will be published by the International Bureau. If the applicant wishes to avoid or postpone publication, a notice of withdrawal of the international application, or of the priority claim, must reach the International Bureau as provided in rules 90 *bis* 1 and 90 *bis* 3, respectively, before the completion of the technical preparations for international publication

Within **19 months** from the priority date, a demand for international preliminary examination must be filed if the applicant wishes to postpone the entry into the national phase until 30 months from the priority date (in some Offices even later).

Within **20 months** from the priority date, the applicant must perform the prescribed acts for entry into the national phase before all designated Offices which have not been elected in the demand or in a later election within 19 months from the priority date or could not be elected because they are not bound by Chapter II.

Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231	Authorized officer THOMAS JOSEPH
Facsimile No. (703) 305-3230	Telephone No. (703) 305-9600

Form PCT/ISA/220 (January 1994)★ *(See notes on accompanying sheet)*

PATENT COOPERATION TREATY

PCT

INTERNATIONAL SEARCH REPORT

(PCT Article 18 and Rules 43 and 44)

Applicant's or agent's file reference PROP98002PCT	FOR FURTHER ACTION	see Notification of Transmittal of International Search Report (Form PCT/ISA/220) as well as, where applicable, item 5 below.
International application No. PCT/US99/24369	International filing date *(day/month/year)* 18 OCTOBER 1999	(Earliest) Priority Date *(day/month/year)* 16 OCTOBER 1998
Applicant PROPHET FINANCIAL SYSTEMS, INC.		

This international search report has been prepared by this International Searching Authority and is transmitted to the applicant according to Article 18. A copy is being transmitted to the International Bureau.

This international search report consists of a total of 4 sheets.

[X] It is also accompanied by a copy of each prior art document cited in this report.

1. [ ] **Certain claims were found unsearchable** (See Box I).

2. [X] **Unity of invention is lacking** (See Box II).

3. [ ] The international application contains disclosure of a **nucleotide and/or amino acid sequence listing** and the international search was carried out on the basis of the sequence listing
   - [ ] filed with the international application.
   - [ ] furnished by the applicant separately from the international application,
     - [ ] but not accompanied by a statement to the effect that it did not include matter going beyond the disclosure in the international application as filed.
   - [ ] transcribed by this Authority.

4. With regard to the **title**,
   - [X] the text is approved as submitted by the applicant.
   - [ ] the text has been established by this Authority to read as follows:

5. With regard to the **abstract**,
   - [ ] the text is approved as submitted by the applicant.
   - [X] the text has been established, according to Rule 38.2(b), by this Authority as it appears in Box III. The applicant may, within one month from the date of mailing of this international search report, submit comments to this Authority.

6. The figure of the **drawings** to be published with the abstract is:

   Figure No. 2
   - [ ] as suggested by the applicant.
   - [ ] because the applicant failed to suggest a figure.
   - [X] because this figure better characterizes the invention.
   - [ ] None of the figures.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24369

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

Group I, claim(s)1-23 and 53-74, drawn to a parameter menu and parameter canvas.
Group II, claim(s) 24 - 41 and 75 - 90, drawn to a preference field.
Group III, claim(s) 42 - 52 and 91 - 120, drawn to a data pallete and data canvas.

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:
1-23 and 53-74

Remark on Protest

☐ The additional search fees were accompanied by the applicant's protest.

☐ No protest accompanied the payment of additional search fees.

International application No.
PCT/US99/24369

Box III TEXT OF THE ABSTRACT (Continuation of item 5 of the first sheet)

This is a graphical user interface which provides for capturing application data in picture form. A set of data capture tools, including a subjective data paramter pallete and accompanying data canvas (215) are presented to an operator during an interactive session. By selecting and placing such data parameters (230) on the data canvas, a user can paint a data picture representing his/her subjective motivations, mental impressions, reasons, etc. for engaging in a particular transaction. In a preferred embodiment, the present interface is used in connection with a stock portfolio management application, which is used to capture user rationale and logic for purchasing or selling financial instruments such as securities, options, etc.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24369

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :G06F 3/14
US CL :345/352

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 345/146, 333-335, 350-354, 356
707/101-104, 906, 908

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,668,996 A (ONO et al) 16 September 1997, especially col. 3, lines 1 - 25.	1-24 and 53-74
A, P	US 5,923,327 A (SMITH et al) 13 July 1999	1-24 and 53 - 74
A, E	US 5,999,177 A (MARTINEZ et al) 07 December 1999	1-24 and 53 - 74

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"B"	earlier document published on or after the international filing date	"Y"	document of *particular relevance*; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 15 FEBRUARY 2000	Date of mailing of the international search report 02 MAR 2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer THOMAS JOSEPH Telephone No. (703) 305-9600

Form PCT/ISA/210 (second sheet)(July 1992)★